OOP

ئامادەکردنی : محمد فایەق کاکە

زانکۆی گەرمیان - زانستی کۆمپیوتەر

2016-2017

پێشەکی

بەناوی خودایی گەورە و دلۆڤان

ئەم بابەتە پێشکەشە بە هەموو ئەو فێرخوازانەی کە ئارەزووی فێربوونی زمانەکانی کۆمپیوتەریان هەیە .

ئامادەکردنی : محمد فایەق کاکە

زانکۆی گەرمیان –زانستی کۆمپیوتەر

Object-Oriented Programming

ئەم بابەتە بابەتێکی هەرە گرنگی زمانی جافایە کە پێکدێت لە کۆمەڵێ بابەتی گرنگ کە پێویستە لەسەر فێرخوازانی زمانەکە بیزانن , پشت بەخوای گەورە لەم کتێبە بچووکەدا باسی تەواوی بابەتەکانی دەکەین .

بابەتەکان پێکدێت لە :

1-Inheritance

2-Overriding

3-Polymorphism

4-Abstraction

5-Encapsulation

6-Interface

7-Packages

پێش باسکردنی ئەو بابەتانەی پەڕەی پێشووتر لەسەرەتادا باسی کۆمەڵێ بابەتی تر دەکەین کە زۆر پێویستە لەسەرەتادا باس بکرێت ..

وەکو .....

1- کلاس

2-میسۆد

3-جۆرەکانی میسۆد

4-ئۆبجێکت

5-جۆرەکانی مۆدیفایەر

6-سترینگ

7-کاراکتەر کلاس

8-فارێبل و جۆرەکانی فارێبل

كلاَس

كلاَس بريتيه له كۆمەڵێ ئۆبجێكت objectكه خاوەنى هەندێك تايبه تمەندى بەخۆيەتى, دەتوانين بڵێن كلاَس بنچينەى ئۆبجێكته.

1–كلاَس ناوكى جافايه هەمووشتێك لەژێردەسەلاتى كلاَس دايه.

2–كلاَس وەسفى هەردوو پێكهاتەى ئۆبجێكت دەكات كەله State و Behavior پێكدێت.

كلاَس پێكدێت له :

1–داتا

2–ميسۆد

3–كۆنسترەكتەر

4–بلۆك

5–ئينتەرفەيس

یاسای ڕاگەیاندنی کلاس لە جافادا :

```
Class ناوی کلاس {
Data;
Method;
}
```

```
class student{
public static void main(String[]args){
System.out.println("hello Wold");
}}
```

1–دەبێ کلاس و فایلی پرۆگرامەکە هەمان ناویان هەبێ.

2–ئەگەر دووکلاس مان هەبووئەگەر بمانەوێ لەناو یەك فایل خەزنی بکەین هەمیشە دەبێ یەکێك لەکلاسەکان public بێ فایلەکەش هەمان ناوی ئەو کلاسە دەبێ.

3-ئەگەر هەردوو کلاسەکەمان public نەبوو فایلەکەمان بەناوی ئەو کلاسە دەبێت کە main()method ی تێدایە .

```
class  student 1{
        }
class student 2 {
public static void main(String[]args) {
          }
```

بانگکردنی کلاس

ئەگەر دوو کلاس مان هەبوو بمانەوێ بانگی میسۆدێك یان فارێبلێك بکەین لەکلاسەکەی تردالە سەرەتا دا ناوی کلاسەکە دواتر (.) دواتر ناوی میسۆد یان فارێبلەکە ,

واباشترە کە میسۆد و فارێبلە بانگکراوەکە public و static بیّ .

بەم شێوەیە ⟸ ; (ئارگویمێنت) ناوی میسۆد . ناوی کلاسی بانگکراو

```
Class  student
public static void main(String[]args){
int result=aa.sum(2,4 );
System.out.println(result);
}
class aa{
public static int sum(int m,int d){
int b=m+d;
return b;
}}
```

ناوی فارێیلی بانگکراو . ناوی کلاسی بانگکراو ;

ئەم یاسایەی سەرەوە بەکاردێ بۆبەکارهێنانی فارێبلێکی دیاری کراو لەناو کلاسێکی تردا .

ئۆبجێکت

هەر شتێك لە دوونیادا بوونی هەبێ و خاوەنی کۆمەڵێ تایبەتمەندی وکرداری تایبەت بەخۆی بێت پێیدەوترێ ئۆبجێکت (object) .

1–هەموو ئۆبجێکتێك (state and behavior)هەیە

2–ئۆبجێکت لە سێ کاراکتەر پێکدێت.

(state ,behavior,identity)

بۆ نموونە سەیارەیەك ئۆبجێکتە

state	behavior
مۆدیل	وەستان
ڕەنگ	گواستنەوە
جۆر	لێخوڕین

وەکو میسۆد بەکاردێ ← Behavior-1

وەکو فارێبل بەکاردێ ← State-2

```
Class Car{
String color;
Public static void drive(){
}}
```

دروستكردنی ئۆبجیكت

بەسێ هەنگاو ئۆبجێکت دروست دەکرێت.

1. Declarationناساندنی فاریێبلێك لەگەڵ ناوی فاریێبڵەکە.

2.Instantiationبەکارهێنانی کیۆردی new

نموونە ← ناوی کلاس ناوی objی = new ناوی کلاس( ) ;

3.Initializationبەکارهێنانی کیۆردی new وەپێدانی نرخ.

نموونە ← ناوی کلاس ناوی obj = new ناوی کلاس(2);

تێبینی //جۆری سێیەم زیاتر لەگەڵ کۆنستراکتەر بەکاردێ .

ڕێگاکانی دروستکرنی ئۆبجێکت

1-By new keyword.

2-By new instance()method.

3-By clone ()method.

4-By factory method.

ڕێگای یەکەم بۆدروستکردنی ئۆبجێکت باشترینە و ئاسانیشە.

لە هەرە سوودەکانی ئۆبجێکت ئەوەیە بۆبانگکرنی میسۆد و فاریێبل بەکاردێ.

ئۆبجێکت Annonymous

anonymous بەزمانێکی سادە واتە بێ ناو ئەم جۆرە ئۆبجێکتە هیچ نیشاندانێکی نیە یان هیچ سەرچاوەیەکی نیە.

ئەگەربمانەوێ یەك جار ئۆبجێکت بەکاربهێنین ئە ئۆبجێکتە باشترینە

بە شێوەی درووستدەکرێ ← new ناوی کلاس;

ئەگەر بانگی میسۆدێك بکەین ← new ناوی کلاس . ناوی میسۆد ( );

```
class  student {
      void fact (int n) {
            int fact=1;
              for(int i=1;i<n;i++) {
                  fact *=I;
              }
      System.out.println("factorial"+fact);
      Public static void main(String[]args){
        new student().fact(5);   ←
            }}
```

چەند ئۆبجێکتێك لەیەك کاتدا

دەتوانین لەیەك کاتدا چەند ئۆبجێکتێك دروست بکەین بەناوی جیاواز

کەپێی دەوترێ Multiple object.

```
class student {
  int length;
  int width;
  void insert(int l,int w) {
  length=l;
  width=w;  }
  void area() {
   system.out.println(length*width); }
  public static void main(String[]args){
   student r1=new student();
   student r2=new student();
   r1.insert(11,5);
   r1.area();
   }}
```

output

55

میسۆد

میسۆد لە جافادا لەکۆمەڵێ statement یان کۆمەڵێ کۆد

بۆ ئەنجام دانی کرداریك پێکەوە کاتێك تۆ بانگی سیستەم

دەکەین ;()system.out.println.

- میسۆد لەجافادا وەسفی پێکهاتەیbehaviorدەکات لەobject دا

واتە دەتوانین behavior بەکاربهێنین بۆدرووستکردنی میسۆد

یاسایی درووستکردنی میسۆد بەگشتی

```
Modifier return type name of method(parameter list)

{ //body method

}
```

modifier ⟸ بریتیە لە کۆمەڵێ کۆد کەچەندین تایبەتمەندی

ئەبەخشێ بە میسۆدەکە

Return type ⟸ جۆری گەڕانەوەی ئەنجامی میسۆدەکە.

ناوی میسۆدەکە ⟵ Name of method

وەرگرتنی نرخی نێردراو ⟵ Parameter list

{ } نوسینی کۆدەکان لەنێوان ⟵ Body method

Public static int sum(int a,int b) { ⟵ نموونە

//Body method

Return; }

Public ,static ⟶ modifier

int ⟶ return type

Sum ⟶ name of method

a,b ⟶ parameter

int a,int b ⟶ parameter list

{ } ⟶ body method

return ⟶ return value

کۆمەڵێ تێبینی لەسەر میسۆد

- مەرج نییە هەمووکات return type لەگەڵ list parameter هەمان data typeهەبێ بۆ نموونە هەردووکیان int بن.

نموونە ⟸ Public static int sum (int a,double b) {

return ;   }

- جۆری return type کەوتوەتە سەر ئەنجامی میسۆدەکە واتە ئەگر ئەنجام پۆینتی لەگەڵ بوو دبێ double بێ.

- ناوی میسۆد هەمیشە بە پیتی سمۆل دەستپێدەکات.

شێوازەکانی نووسینی میسۆد

1-public static void sum() { }

2-public static void sum(parameter list) { }

3-public static int sum(parameter list) { return ; }

ئەکرێ جۆرێکی تری return type بێ

جیاوازی نێوان void و return type

- لە void دا ئەنجامی میسۆدەکە دەگەڕێتەوە بەڵام ناتوانین بەکاری بهێنینەوە دووبارە ✗ ژمارەیەك + ئەنجامی میسۆد ;
- لە return type کاندا ئەنجامی میسۆدەکە دەگەڕێتەوە ودەتوانین دووبارە بەکاری بهێنینەوە ← ژمارەیەك + ئەنجامی میسۆد ;

```
public class mu {
public static void main(String[]args)  {
int operation = sum(2,4);
int result = operation +4;      ← بەکارهێنانەوە
system.out.println(result);
  }
Public static int sum(int a,int b) {
 int A = a+b;
 return A;
  }
```

output

10

بانگکردن لە میسۆدا

دەتوانین لەناو جافادا بانگی میسۆد بکەین جا لە ناو هەمان کلاَسدا بێت یان لەناو کلاسێکی دیکەدا بێت .

دوو جۆر بانگکردنمان هەیە :

1ـ میسۆد بەگەڕانەوەی نرخی میسۆد ودووبارە بەکارهێنانەوەی .

ناوی میسۆدەکە لەناوsysدەنوسرێت و لەناو main methodدا

یان دەتوانین ناوی میسۆدە بانگکراوەکە لەناو فارێبڵ دابنێین دواتر ناوی فارێبڵەکە لەناو سیستەمەکە ئەنووسین,ئەم میسۆدە return typeهەیە.

نموونە

```
int result = ناوی میسۆد ( نرخ )
System.out.println(result);
```

2-میسۆد بەگەڕانەوەی نرخ بەلاَم ناتوانین دووبارە بەکاری بهێنینەوە.

ئەم میسۆدە کیۆردی void لەگەڵ نرخی میسۆدەکە دەگەڕێتەوە بەلاَم ناتوانین جارێکی تر بەکاری بهێنینەوە,دەبێ سیستەمەکە لەناو میسۆدە بانگکراوەکە بنووسین بەلاَم ناوی میسۆد هەمووکات لەناو main method دەنووسرێت .

نموونە

```
Public static void sum(int a,int b) {
Int result = a+b;
}
```

جیاوازی نێوان parameter list و Argument

Argument-1 بریتیە لەونرخی کەبۆ میسوۆدی بانگکراو دەنێردرێت,

parameter list-2 ئەرکی وەرگرتنی نرخەکەیەو لەکاتی بانگکردنی میسوۆدەکەدا درووست دەکرێت.

```
public class student  {
    public static void main(String[]args) {
        sum(2,6) ;
    } // argument      parameterlist
public static void sum(int a,int b){
        int  f= a+b;
        }}
```

دوو جۆری سەرەکی میسوۆد

دوو جۆری سەرەکی میسوۆد هەیە:

static method-1واتە کیوۆردی staticی لەگەڵە.

non-static method-2 واتە کیوۆردی staticی لەگەڵ نییە.

Modifier مۆدیفایەرەکان

مۆدیفایەرەکان بریتین لە کۆمەڵێ وشە کەدەچنە سەر میسۆد و فارێبل و کلاَسەکان چەندین تایبەتمەندیان پێدەبەخشن.

دوو جۆر مۆدیفایەر هەیە

1-Java Access Modifier

2-Non Access Modifier

چوار جۆر وشە (کیۆرد)مان هەیە لە access modifier

1-Default (no keyword)

2-Private

3-Public

4-Protected

کیۆردی Default

بریتیە لەو کیۆردەی لەناوئەوکلاسی پاکێجێك دا هەی بەلاّم بەهیچ شێوەیەك دەرناکەوێ واتە:ئەگەر کلاس یان میسوۆدمان بینی هیچ کیۆردێکی لەگەڵ نەبوو ئەوا کیۆردی Default بەکارهاتووە بەلاّم دەرنەکەوتووە.

```
String version="student";
 boolean  processor() {
  return true;
   }
```

کیۆرد بەکارنەهاتووە

کیۆردی Private

ئەم کیۆردە تەنها لەناو کلاس دا هەیە و لەگەڵ میسوۆد وفاریێبل وکۆنستراکتەردا بەکاردێت .

ئەم کیۆردە بڕواتەسەر میسوۆد یان فاریێبلێك لەناو کلاسێك دا ناتوانین جاریکی تر ئەم میسوۆد وفاریێبلە بەکاربهێنینەوە لە کلاسێکی تردا.

نموونەکەی لەپەڕەکەی دیکەیە

class a { ← نموونەیەك لەسەر كیوۆردی private

```
private int data = 20 ;
private void msg () {
System.out.println("hello java");
 } }
 public class b {
Public static void main(String[]args) {
   a   opj = new  a;
System.out.println(obj.data);
obj.msg();
  } }
```

Output

Compile error

Output ئیرۆر دەبیٚ چونكە بانگی میسوٚدێك و فارێبلێكمان كردووە

بۆناو كلاسێكی دیكە كە كیوۆردی private بەكارهێناوە.

کیوۆردی public

ئەم کیوۆردە دەچێتە سەر (کلاس,میسوۆد,فارێبل,ئینتەرفەیس) .

ئەم کیوۆردە مانای گشتی دەدات واتە لەهەر کوێ پێویستمان بەم کلاس و فارێبل و میسوۆدە هەبوو دەتوانی بەکاری بهێنیت لە هەرکوی بتەوێت.

```
class a {
public int data =33;
 public void msg() {
System.out.println("java");
 } }
public class mu {
Public static void main(String[] args) {
 a  obj = new  a( );
System.out.println(obj.data);
 obj . msg();
  } }
```

Output

33
java

لەبەر ئەوەی کیوۆردی public بەکارهاتووە لەگەڵ میسوۆد و فارێبڵەکەدا دەتوانین لەهەرکوێ بمانەوێ بەکاری بهێنین.

کیۆردی protected

ئەم کیۆردە لەگەڵ فارێبل ومیسۆد وکۆنستراکتەردابەکاردێت لەناوsuperclass یان subclassەکە لەناو پاکێجێك دا یان لەناو کلاًسێکدا لەناو پاکێجێك دا.

- Superclassدواتر لەبابەتی inheritance دا باسی دەکەین.
- لەڕێگەی ئەم کیۆردەوە دەتوانین ئەو میسۆد وفارێبلأنەی کەوا لەناو superclass دان بتوانین لە subclass دابەکاریان بێنین.
- ئەم کیۆردە لەگەڵ کیۆردی class و interface دا بەکارنایەت تەنانەت لەگەڵ ئەومیسۆد وفارێبلأنەش کەدەکەونە ناو interface.

```
public class a {
Protected int data=14;
  }
```

Super class

```
class b extends a {
 Public static void main(String[]args) {
    b  obj = new b();
   System.out.println(b.data);
    } }
```

subclass

output

14

Non Access مۆدیفایەری

ئەم جۆرە مۆدیفایەرە لە چەند کیۆردێك پێکدێت :

1-static modifier

2-final modifier

3-abstract modifier

4-synchronized and volatile modifier

تێبینی// ئێمە لێرەدا باسی کیۆردی synchronized وvolatile ناکەین

چونکە لە OOP دا بەکارنایەت.

محمد فایەق کاکە

زانکۆی گەرمیان – زانستی کۆمپیوتەر

کیۆردی Static

ئەم کیۆردە بەکاردێ لەگەڵ میسوّد وفارێبل وبلۆك ونێستدکلاّس.

فارێبلی static

کیۆردی static لەسەر فارێبل وا دەکات میمۆری پرۆگرامەکەی تۆچالاك تربێت.

فارێبلی static یەکجار شوێن دیاری دەکات لەناو میمۆری.

فارێبلی static دەتوانین ڕاستەوخۆ بانگی بکەین بەبێ درووستکردنی ئوبجێکت.

فارێبلی static زۆر زیاتر ڕێکخراوترە.

```
class mu {
 int count = 0 ;
 mu ( ) {
  count++;
 System.out.println(count);   }
 public static void main(String[]args) {
 mu  obj = new mu();
 mu  obj = new mu();
 mu  obj = new mu();
 } }
```

static بەکارنەهاتووە

output

لەگەڵ static
1
2
3

output

بەبێ static
1
1
1

میسۆدی static

میسۆدی static واتە ئەو میسۆدەی کیوۆردی static بەکارهێناوە.

میسۆدی static بۆبانگکردنی پێویستی بەدروستکردنی ئۆبجێکت ناکات.

ئەگەر میسۆدەکەمان static بوو دەبێ فارێبلەکان ناوی static بی.

لە میسۆدیstatic دا دەتوانین نرخی داتاکانمان بگۆڕین.

```
class calculator {
static int operation (int x ) {
return x*x*x;
    }
public static void main(String[]args) {
 Int result =operation(5);
  System.out.println(result);
   } }
```

output

125

کیۆردی Final

- ئەم کیۆردە لەجۆری non access modifier ە.
- ئەم کیۆردە لەگەڵ میسۆد و فارێبل وکلاّسەکان دا بەکاردێت .
- ئەم کیۆردە کاتێك دەڕواتە سەر فارێبڵەکان نەگۆڕیان دەکات واتە ئەو

ئەو داتایەی لەناو فارێبڵ و دایە جارێکی تر ناتوانین بیگۆڕین بەلاّم

بەمەرجێك لەگەڵ کیۆردی static دابێت.

- کاتێك دڕواتەسەرکلاّس ناتوانین کرداری inheritance ئەنجامبدا.
- کاتێك ئەم کیۆردە دەڕواتە سەر میسۆد ئیتر ئەو میسۆدە ناتوانیٚ کرداری

Overridden ئەنجام بدات دواتر باسی overridden دەکەین.

```
Public class test {
Public static final int DATA = 13;
Public static void main(String[]args) {
Data =16;
System.out.println(Data);
}}
```

output

error

Output ئیرۆر دەبیٚ چونکەفارێبڵەکەمان final ئەو دواتر نرخی فارێبلّمان گۆڕیوە

کیۆردی abstract

.

کیۆردی abstract لەگەڵ کلاس و میسۆدەکاندا بەکاردێت.

کلاسی abstract

کلاسی abstract واتە ئەوە کلاسەی کە کیۆردی abstract بەکاردێنێ.

کلاسی abstract دوو سوودی گرنگی هەیە:

1– بەنهێنی هێشتنەوەی شێوازی ئەنجامدانی کاری کلاسەکەتەنها شیوازەکە بە بەکارهێنەر پیشاندەدات وەکو کرداری ناردنی نامە.

2– ئەنجامدانی کرداری inheritance.

abstract class a{ } ⟵ شێوازی ڕاگەیاندنی

```
abstract class a{
 void mag()
    }}
class b extends a {
public static void main (String[]args){
  obj =new b();
   obj.msg();
     } }
```

میسۆدی abstract

ئەگەر میسۆد کیۆردی **abstract**ی بەکارهێناپێیدەوترێت میسۆدabstract.

میسۆدی abstract براکێتی نییە { } و لەگەڵ کیۆردی finalدا بەکارنایەت.

```
abstract class a {
  abstract void msg();
  }
 class b extends a {
 void msg () {
      } }
```

شێوازی ڕاگەیاندنی (abstract void msg();)

- میسۆدی abstract یارمەتی کلاسی abstrac دەدات بۆئەنجامدانی کرداری Inheritance(میرات) لەناوsuperclass و subclassدا بەکاردێت .
- ئەگەر میسۆد abstract بوو دەبێت کلاسەکەش abstract
- لەمیسۆدی abstract دابراکێت بەکارنایەت وتەنها سیمیکۆلۆن(;) دادەنرێت .

تێبینی // ئەگەر میسۆدەکەمان کیۆردی abstract لەگەڵ بوو دەبێ کلاسەکەش لەگەڵی بێت بەلام مەرج نیە هەموو کلاسێکیabstract میسۆدەکەشی abstract بێ.

Identifiers ڕێساکانی ناونان

هەموو بەشەکانی جافا پێویستیان بە ناونان هەیە ,ئەم ناوانە بەکاردێ بۆ کلاْس و فارێبل و میسۆد کە پێیدەوترێ Identifiers.

کۆمەڵێ خاڵی گرنگ هەیە پێویستدەکات زۆرکات لەبیرت بێت :

هەموو identifier پێوێستە دەست پێبکات بە پیتی(A بۆ Z یان a بۆ z) وە دەتوانین هێمای دراوی دۆلار ($)و underscore (_) بەکاربهێنین.

```
String 123abc;   ✗
String $abc;     ✓
String abc_wrr;  ✓
String ab.cd ;   ✗
```

بەهیچ شێوەیەك ناتوانین کیوۆردەکان لە گەڵ فاریبڵەکان بەکاربێنین بەلاْم دەتوانین کەمێك کەم وزیادی پیتەکانی کیوۆردەکەبکەین یان بچووك و گەورەی بکەین.

```
int class;    ✗
int clas ;    ✓
int Class;    ✓
```

فارێبڵ Variables

فارێبڵ لەجافادا واتە ئەو شوێنەی کە داتاو زانیاری تێدا هەڵدەگیرێ.

هەموو فارێبڵێك data type تایبەت بەخۆی هەیە کە گونجاوە لەگەڵیدا.

- ➢ فارێبڵەکان بەچەندین شێواز ڕادەگەیەنرێت و نرخی پێدەدرێت.

شێوازی ڕاگەیاندن

```
1-int a,b,c;
2-int a;
```

شێوازی ڕاگەیاندن و نرخ

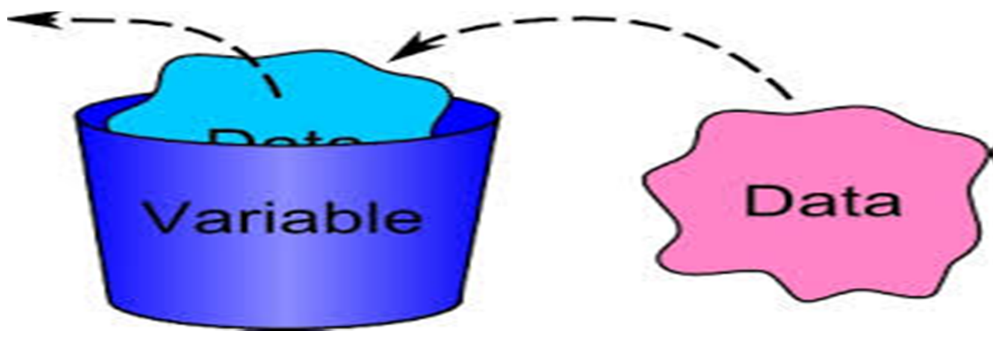
```
1-int a=2,b=4,c=5;
2-int a=23:
```

لەجافادا دوو شێوازی دروستکردنی فارێبڵ هەیە:

1–تەنها ڕادەگەیەنرێت و نرخی پێنادرێ وەکو نموونەی سەرەوە.

2–ڕادەگەیەنرێت و نرخیشی پێدەدرێت وەکو نموونەی سەرەوە.

تێبینی// هەمووفارێبڵێك بەلایەنی کەم پێش بەکارهێنان دەبێ ڕابگەیەنرێت.



سێ جۆر فارێیلمان هەیە

1-local variable

2-instanance variable

3-static/class variable

Local variable

- Local variable لەناومیسۆد وکۆنستراکتەر و یان بلۆك ڕادەگەیەنرێت.
- Local variab كاتی دروست دەكرێ كەمیسۆد یان كۆنستراكتەر یان بلۆك درووست دەكرێ كاتێكیش نامێنێ كە میسود یان كۆنستراكتەر یان بلۆك كوتایی پێدێت .
- Access modifier كان لەگەڵ Local vaiable بەكارنایەت.

Public static int d =12; ← نموونە

- Loacal variable پێش بەكارهێنانی دەبێ نرخی پێبدرێت واتە حاڵەتی Default value لەم جۆرە فارێبلەدا ڕوونادات.

```
Public static void mag() {
    int d = 15;
```

← پێش بەكارهێان نرخی پێدەدرێت

Instance variable

- Instance variable لەناو کلاّس رادەگەیەنرێت بەلاّم لەدەرەوەی میسۆد وکۆنستراکتەر یان بلۆك .
- Instance variable کاتێك دروست دەکرێت کە ئۆبجێکت دروست دەکرێت کاتێك ئۆبجێکت نامێنێ instance variable یش نامێنێ.
- بەهۆی ئۆبجێکتەوە دەتوانین instance variable بانگی بکەینە ناو main method ودواتر بەکاری بهێنین.
- Instance variable ڕادەگەیەنرێت لەناو کلاّس دا پێش بەکارهێنان دوای بەکارهێنان .
- لە instance variable دا دەتوانین Access modifier بەکاربهێنین.
- دەتوانین instance variable بەبێ پێدانی نرخ بەکاربهێنین بەلاّم

بەلاّم حاڵەتی default value ڕوودەدات دواترلەکۆنستراکتەرباسی دەکەین.

```
class stusent {
public string =”zankoy garmyan”;
public static void  mag( ) {
        } }
```

Static /class variable

- ئەم فارێبڵە static variable یان class variable پێدەوترێت.
- ئەم فارێبڵە لەناو کلاس درووست دەکرێت بەبەکارهێنانی کیۆردی بەڵام لەدەرەوەی میسۆد یان کۆنستراکتەر یان بلۆك.
- Static variable بەدەگمەن نرخی نەگۆڕە بەتایبەتی ئەگەر لەگەڵ کیۆردی final یان کیۆردی private هات.
- Static variable درووست دەکرێت کاتێك programدروست دەکرێت کۆتایشی پێدێت کاتێك program ەکە کۆتایی دێت یان دەوەستێ.
- Static variable دەتوانین ڕاستەوخۆ بانگی بکەین بەبێ درووستکردنی ئۆبجێکت.
- دەتوانین لەدوو شوێنی جیاواز ڕاپبگەیەنین و نرخی پێبدەین.
- بەهەمان شێوەیinstance variable حاڵەتی default value لە static variable ڕوودەدات دوتر لە کۆنستراکتەر باسی دەکەین.

```
public class student {
private static int salary ;
public static final String department =”davalopment”;
public static void main( String[]args)
salary = 1000;
System.out.println(slary+department);
}}
```

output

davalopment

100

Character Class

کاراکتەر کلاًسەکان بەشێوەیەکی ئاسایی کار لەگەڵ داتا تایپی charدەکەن.

بە دوو ڕێگە دەتوانین کاراکتەر کلاًس و کاراکتەرمیسوّدەکان بەکاربهێنین.

1– بەکارهێنانی char literal ⟵ char m = ‘d’;

2–دروستکردنی ئۆبجێکتیCharacter ⟵ Character f=new Character();

ئەوەی گرنگ بێت لێرەدا باسکردنی کومەڵێ میسوّدی ئامادەکراوە.

کاراکتەر میسوّدەکان

1-isLetter()

2-isDigit()

3-isWhitespace()

4-isUpperCase()

5-isLowerCase()

6-toUpperCase()

7-toLowerCase()

8-toString()

بۆ هەموویان دەبیٚ ئۆبجێکتی character درووست بکەین.

یان پێش میسودەکە دەنووسین character دواتر . دواتر ناوی میسوّدەکە

isLetter () كاراكتەر میسۆدی

ئەم میسۆدە بەکاردێ بۆئەوەی بزانین ئایا نرخەکەمان پیتە, ئەگەر پیت بوو

ئەوا ئەنجام true دەبێت ئەگەر وانەبێت false دەبێت.

یاسایی دروستکردنی ← boolean isDigit (char m)

```
class mu {
public static void main(String[] args) {
System.out.println(Character.isLetter('c'));
System.out.println(Character.isLetter('5'));
}}
```

output

true
false

isDigit() كاراكتەر میسۆدی

ئەم میسۆدەبەکاردێ بۆئەوەی دیاری بکەین ئایا نرخەکەمان ژمارەیە.

ئەنجامی میسۆدەکە true دەبێ ئەگەر نرخەکە ژمارە بێت, false دەبێ

ئەگەر نرخەکە ژمارە نەبێت.

یاسای درووستکردنی ← Booleab isDigit (char m)

```
class  mu {
Public static void main(String[]args){
System.out.println(Character.isDigit('a'));
System.out.println(Character.isDigit('5'));
}}
```

output

false
true

کاراکتەر میسۆدی isDigit( )

ئەم میسۆدە بەکاردێ بۆ دیاریکردنی '_' ,\t, \n ئایا نرخەکە بۆشایی تێدایە.

ئەنجامەکەtrueدەبێ ئەگەر نرخەکەبوش بێت, ئەگەر بۆش نەبوو falseدەبێ

یاسایی دروستکردنی ← boolean isWhitespace(char m)

```
class mu {
 Public static void main(String[]args) {
System.out.println(Character.isWhitespace('a'));
System.out.println(Character.isWhitespace(' '));
System.out.println(Character.isWhitespace('\n'));
 } }
```

output

false
true
true

کاراکتەر میسۆدی isUpperCase()

ئەم میسۆدە بەکاردێ بۆدیاریکردنی ئایا نرخەکە پیتی گەورەیە.

ئەگەر نرخەکە پیتی گەورە بوو ئەنجام true دەبێ ئەگەر وانەبوو ئەنجامەکە false دەبێت.

یاسایی درووستکردنی ← boolean isUpperCase (char m)

isUpperCase نموونە

```
class mu {
public static void main(String[]args) {
System.out.println(Character.isUpperCase ('c'));
System.out.println(Character.isUpperCase('C'));
 } }
```

output

false
true

کاراکتەر میسۆدی isLowerCase

ئەم میسۆدە بەکاردێ بۆدیاریکردنی ئایا نرخەکە پیتی بچووکە ئەگەرنرخەکە پیتی بچووك بوو ئەنجام true دەبێ ئەگەر وانەبوو ئەنجامەکەfalse دەبێ.

یاسایی درووستکردنی boolean isLowerCase()

```
class a {
public static void main( String[]args){
System.out.println(Character.isLowerCase('c'));
System.out.println(Character.isLowerCase('C'));
}}
```

output

true
false

toUpperCase() کاراکتەر میسۆدی

ئەم میسۆدە بەکاردێت بوگەڕانەوی پیتەکە بە پیتی گەورە.

boolean toUpperCase(char m) ⟵ یاسایی درووستکردنی

```
class mu {
 public static void main(String[]args) {
System.out.println(Character.toUpperCase('a'));
System.out.println(Character.isLowerCase('C'));
 } }
```

output

A
C

toLowerCase کاراکتەر میسۆدی

ئەم میسۆدە بەکاردێت بوگەڕانەوی نرخەکە بە پیتی بچووک.

boolean toLowerCase(char m) ⟵ یاسایی درووستکردنی

```
class m {
 public static void main(String[]args) {
 System.out.println(Character.toUpperCase('a'));
 System.out.println(Character.toUpperCase('H'));
 } }
```

output

a
a

کاراکتەر میسۆدی toString

ئەم میسۆدە بەکاردێ بۆگەڕانەوەی پیتەکە وەکو خۆی ئەگەر نرخەکە گەورە بوو گەورە دەگەڕێتەوە ئەگەر بچووک بوو بچووک دەگەڕێتەوە.

یاسایی درووستکردنی ← boolean toString(char m)

```
class mhamad {
public static void main(String[]args) {
Character  obj = new Character ('a');
System.out.println(obj.toString());
System.out.println(Character.toString('H'));
}}
```

output
a
H

تیبینی// لەم پرۆگرامەدا هەردوو شێوازەکەم بەکارهێناوە.

OOP بەکوردی

محمد فایەق

زانستی کۆمپیوتەر– زانکۆی گەرمیان

سترینگ String

سترینگ لەجافادا پێکدێت لە کۆمەڵێ کاراکتەری یەك لەدوای یەك .

سترینگەکان لەجافادا دەتوانن مامەڵە بەکەن لە گەڵ ئۆبجێکت دا.

دەتوانین سترینگەکان بەکاربهێنین ئەویش بەدوو رێگا .

1-String literal → String m="zankoy garmyan";

2-by using object → String m= new String("test");

نرخی سترینگەکان ناگۆڕێن بەلاَم ئەگەر ویستمان گۆڕانکاری لەنرخەکاندا بکەین

دەتوانین میسۆدی ئامادەککراوی جافا بەکاربهێنین ئەم میسۆدانەش دەبنە

دوو بەشەوە:

1-String Buffer

2-String Builder

هەریەك لەم دووبەشەی سترینگە لەکۆمەڵێ میسۆدی ئامادەککراو پێکدێت.

تێبینی // کاتێك ئەم دووبەشە بەکاردێنین گەر بمانەوێ گۆڕانکاری لە نرخی

سترینگەکە بکەین.

String Buffer

ئەم بەشەی سترینگە پێکدێت لەکۆمەڵێ میسۆدی ئامادەکراو:

1-append

2-reverse ⟵ بۆ پێچەوانەکردنەوەی نرخەکە بەکاردێ

3-delete ⟵ بۆسرینەوەی نرخەکە لەماوەیەکی دیاریکراو کەخۆمان دیاری دەکەین

4-insert ⟵ بۆ داخل کردنی نرخێك بۆ سەرنرخی سترینگەکەمان

5-replace

---

append

ئەم میسۆدە بەکاردێ بۆخستنەسەری نرخێك بۆسەر نرخێکی تر.

ئەم میسۆدە لەگەڵ(data typeint,double,float,char,string)بەکاردێ

append (boolean m) ⟵ یاسای درووستکردنی

append (int m)

append (string m)

append (char m)

```
class mu {
public static void main(String [] args){
StringBuffer  obj = new StringBuffer("zankoy");
obj.append("garmyan");
    System.out.println(obj);
        }}
```

output

Zankoygarmyan

length() میسۆدی

ئەم سترینگە بەکاردێ بۆ دیاریکردنی ژمارەی ئیندێکسەکانی نرخی سترینگەکە.

```
class mu  {
Public static void main(String [] args ) {
String   result =”zankoy garmyan”;
int  len =result.length();
System.out.println(len);
} }
```

output

13

concat ( ) میسۆدی

ئەم سترینگە بەکاردێ بۆ پیشاندانی نرخی دوو سترینگ پێکەوە.

```
String 1 .concat(string 2);
```

یاسای دروستکردنی

```
class mu {
public static void main(String []args) {
String d=”zankoy”;
String f=”garmyan”;
String  result = d.concat(f);
System.out.println(result);
} }
```

output

zankoygarmyan

میسۆدی charAt( )

ئەم سترینگە بەکاردێت بۆ هێنانی نرخی ئیندێکسێکی دیاری کراو لە سترینگدا.

charAt(int idex); ⟵ یاسای درووستکردنی

```
class mu {
public static void main(String []args) {
String  str = "Kurdistan";
System.out.println(str.charAt(3));
} }
```

output

K

میسۆدی contentEquals()

ئەم سترینگە بەکاردێ بۆبەراوردکرنی نرخی دووسترینگ ئەگەر یەکسانن بوون ئەوا ئەنجام true دەبێ وە ئەگەر یەکسان نەبوو ئەنجام false دەبێ.

boolean cotentEquals (strint str) ⟵ یاسای دروستکردنی

```
Class  mu {
Public static void main(String[]args) {
String  str1="zankoy garmyan";
String str2 = "zankoy hawler";
StringBuffer  str3 =new  StringBuffer("zankoy garmyan");
boolean  result =str1.cotentEquals(str3);
System.out.println(result);
result = str2.contentEquals(str3);
System.out.println(result);
} }
```

output

true

false

میسۆدی copyValueOf()

ئەم میسۆدە بەکاردێ بۆ گۆڕینی نرخی Character بۆ String.

یاساى دروستکردنی ← String copyValueOf(char[]ch)

```
class mu {
Public static void main(String[]args) {
char [] ch ={'g','a','r','m','y','a','n'};
String  str = " ";
str =str .copyValueOf(ch);
System.out.println(str);
        }}
```

output

garmyan

میسۆدی copyValueOf()

ئەم میسۆدەش هەمان کرداری میسۆدی پێشوو ئەنجام دەدات بەلام لێرەدا

ماوەی گۆڕینی نرخی Character بۆ String دیاری بکەین.

یاساى دروستکردنی ← copyValueOf(char[]data,int start,int end)

```
   class  mu {
 public static void main(String[]args) {
char[] data ={'k','u','r','d','i','s','t','a','n'};
 String str =null;
 str =str .copyValueOf(data,1,6);
 System.out.println(str);
            }}
```

له ئیندێکسی 1 بۆ 6 پیشاندەدات ← str =str .copyValueOf(data,1,6);

output

urdist

میسۆدی endsWith()

ئەم میسۆدە بەکاردێ بۆ ئەوەی بزانین ئایا نرخی سترینگەکە بەم نرخە کۆتایی.

ئەگەربەوەکۆتای هاتبوو کەپێی دەلێین ئەوائەنجام true دەبێ گەروانەبوو Falseدەبێ.

یاسای درووستکردنی ← boolean endsWith(String str)

```
classs mu {
public static void main (String []args) {
String str = new String ("garmyan");
boolean result ;
result = str.endsWith("yan");
System.out.println(result);
result =str . endsWith("gar");
System.out.println(result);
}}
```

output

true
false

میسۆدی equals()

ئەم میسۆدە بەکاردێ بۆئەوەی بزانین نرخی دووسترینگ یەکسانن بەیەکتری.

ئەگەرنرخی سترینگەکانمان یەکسان بوون ئەوا ئەنجام true دەبێ یەکسانیش نەبنfalseدەبێ.

یاسای دروستکردنی ← boolean equals(str)

نموونەیەك لەسەر میسۆدی equals

```
classs  mu {
public static voia main (String[]args){
String  str1 =new String(“kalar”);
String  str2 =str1
String  str3 =new  String (“sulaymani”);
boolean  result;
result =str1 .equals(str2);
System.out.println(result);
result =str2 .equals(str3);
System.out.println(result);
}}
```

output

true
false

میسۆدی equalsIgnoreCase()

ئەم میسۆدە هەمان ئیشی میسۆدی پێشووتر دەكات بەڵام ئەم میسۆدە گرنگی

بە پیتی گەورە و بچووكی نادات لەنێو نرخی سترینگە تەنها یەكسان بن .

```
classs mu {
public static void main(String[]args) {
String  str1 =”garmayan”;
String  str2 = “GARMYAN”;
boolean  result =str1 . equalsIgnoreCase(str2);
System.out.println(result);

}}
```

output

true

میسۆدی getChars()

میسۆدەبەکاردێ بۆگۆڕینی نرخی سترینگ بۆCharacterلەشوێنێکی دیاریکراو

یاساى درووستکردنی getChars(int start,int end,char[]ch , destenation begin)

```
class  mu {
public static void main(String[]args) {
String  str1 ="welcome to this tutorial";
char []str2 =new char[7];
str1 .getChars(2,9,ch,0);
System.out.println(ch);
}}
```

واتە 7کاراکتەر دەربێنە ← `char []str2 =new char[7];`

output

7کاراکتەرە → lcom to

میسۆدی indexOf(int ch)

ئەم میسۆدە بەکاردێت بۆ دیاریکردنی ئیندێکسی کاراکتەرێك.

یاسای دروستکردنی → `int  indexOf(int ch)`

```
class  mu {
public static void main(String[]args){
String str = "peshmarga";
System.out.println(str . indexOf('h'));
}}
```

output

3

ئەگەر نرخەکە h تێدا نەبوو ئەنجام 1− دەبێ

indexOf (int ch ,int fromi ndex) میسۆدی

ئەم میسۆدە هەمان کاری میسودی پێشووتر دەکات میسۆدەکە بەکاردێ بۆ دیاریکردنی ئیندێکسی کاراکتەرێك بەلام خۆمان دیاریدەکەین لەکوێوەبەدوایدا بگەڕێت.

int indexOf(int ch ,intfrom index) ← یاسای دروستکردنی

```
class mu {
public static void main(String[]args) {
String str = "kalar Kurdistan";
System.out.println(str.indexOf('s',4));
}}
```

واتەلەئیندیکسی4 بەدواوە بەدوای ئیندێکسی s بگەڕێ

output

11

ئەگەر نرخەکە S تێدانەبووئەنجام 1– دەبێ

---

indexOf(String str) میسۆدی

ئەم میسۆدە بەکاردێ بۆ گەڕان بەدوای ئیندێکسی ستریینگێکدا بۆ ئەوەی بزانین ئیندێکسی ئەو میسۆدە لە کوێوە دەست پێدەکات.

String (String str) ← یاسای دروستکردنی

```
class mu {
Public static void main (String []arsg) {
String str1 ="bzhy peshhmargay kuurdistan";
String str2 ="peshmargay";
System.out.println(str1 . indexOf(str2);
}}
```

نرخیstr1 لەئیندێکسی چەنەوەدەستپێدەکات لەناونرخی str2

output

5

ئەگەر کارەکە نەگونجا ئەنجام 1– دەبێ

indexOf(String str,int fromIndex) میسۆدی

ئەم میسۆدە بەکاردی بۆدیاری کردنی ئیندێکسی سترینگێك بەلاَم خۆمان دیاریدەکەین لە کوێوە بەدوایدا بگەڕێت.

```
class mu {
public static void main (String []args) {
String str1 =”kalar garmyan Kurdistan”;
String str2 =”Kurdistan”;
System.out.println(str1.indexOf(str2,12));
}}
```

لەئیندێکسی12بەدواوەبەدوای ئیندێکسیstr2بگەڕێ لەناوstr1

ئەگەرکارەکەنەگونجا یان کەموکورتی لەنرخیstr1 هەبوو ئەنجام 1– دەبێ.

output

14

lastIndexOf(int ch) میسۆدی

ئەم میسۆدە بەکاردێ بۆدیاریکرنی ئیندێکسی کۆتا کاراکتەریکی دیاریکراو. لەناو نرخی سترینگێك.

```
class mu {
public static void main(String []args) {
String str = “welcome to tutorialspoint.com”;
System.out.println(str.lastIndexOf(‘o’));
}}
```

واتە کۆتا O ئیندێکسی چەندە

output

27

ئەگەر str پیتی O تێدانەبوو ئەنجام 1– دەبێ

lastIndexOf(int ch,int fromIndex) میسۆدی

ئەم میسۆدە بەکاردێ بۆگەڕان بەدوا ئیندێکسی کاراکتەرێك لەناو نرخی

سترینگێك بەڵام خۆمان دیاری دەکەین تاوەکو کوێ بەدوایدا بگەڕێت .

```
class  mu {
Public static void main(String[]args) {
String  str =”welcome to tutorialspoint”;
System.out.println(str.lastIndexOf(‘o’,10));
}}
```

واتە تاوەکو ئیندێکسی 10 بەدوای کۆتا ئیندێکسیO دا بگەڕێ

output

8

ئەگەر نرخەکە تاوەکو ماوەکە O تیدانەبوو ئەوا ئەنجام 1− دەبێ

تیبینی // ئەم میسۆدە جیاواترە لە میسۆدی indexOf(int ch,int fromIndex)

لە میسۆدی indexOf(int ch, int fromIndex) ئەو ماوەیەی دیاری دەکەین ئیتر لەو شوێنەوە دەست بەگەڕان دەکات بەلام لە میسۆدی lastIndexOf(int ch,int fromIndex) ئەوماوەیەی دیاری دەکەین تەنها هەتاوەکو ئەوە ماوەیە بەدوایدا دەگەڕێت.

1-lastIndexOf(String str)

2-lastIndexOf(String str,int fromIndex)

تیبینی// لێرەدا باسی ئەم دوو میسۆدە ناکەین هەمان شێوازی میسۆدەیەکانی پێشتر بەکاردێنین بۆ دۆزینەوەی ئیندێکسی سترینگێك.

میسۆدی replace()

ئەمیسۆدە بەکاردێ بۆگۆڕینی شوێنی دووکاراکتەر لەناو نرخی ستریگێك.

String replace (char oldChar , char newChar) ⟸ یاسای درووستکردنی

```
class mu {
public static void main(String[]args) {
String str = "Kurdistan";
System.out.println(str .replace('k','n'));
}}
```

output

nurdistak

میسۆد replaceAll()

ئەم میسۆدە بەکاردێ بۆ گۆڕینی هەموونرخی سرینگەکە بۆ نرخێکی دیکە.

String replaceAll (String str ,replacement) ⟸ یاسای دروستکردنی

```
class mu {
public static void main(String[]args) {
String str ="kalar Kurdistan";
System.out.println(str.replaceAll("kalar Kurdistan","peshmarga"));
}}
```

output

peshmarga

میسۆدی replaceFirst()

ئەم میسۆدە بەکاردێ بۆ ئاڵوگۆڕکردنی سەرەتای نرخی سترینگەکەمان لەگەڵ نرخێکی دیکەدا.

یاسای درووستکردنی ⬅ String replaceFirst(String str ,String replacement)

```
Class  mu {
Public static void main(String[]args) {
String str =kalar Kurdistan";
System.out.println(str.replaceFirst("kalar","Sulaymani"));
}}
```

output

Sulaymani kurdistan

میسۆدی split()

ئەم میسۆدە بەکاردێ بۆبەش بەش کردنی نرخی سترینگەکە.

یاسای دروستکردنی ⬅ String[]split(String str,int limit)

یان ⬅ String[]split(String str)

```
class mu {
Public static void main(String[]args) {
String str = "kalar – Kurdistan – karkuk";
for (String result : str.split("_",3)) {
System.out.println(result);
}}}
```

واتە بیکە 3 سێ بەشەوە

output

kalar
kurdistan
karkuk

میسۆدی startsWith()

ئەم میسۆدە بەکاردێ بۆ ئەوەی بزانین نرخی سترینگەکەمان بەم نرخە دەست پێدەکات کەخۆمان پێدەڵین.

ئەگەر بەو نرخە دەستی پێکردبوو ئەوا true دەبێ ئەگەر وانەبوو false دەبێ.

String startsWith(String str, int index) ← یاسای دروستکردنی

String startsWith(String str) ← یان

```
class mu {
public static void main(String[]args) {
String str =”welcome to this tutorials”;
System.out.println(str.startsWith(“welcome”));
System.out.println(str.startsWith(“this”));
System.out.println(str.startsWith(“this”,11));
}}
```

واتە وشەی this لە ئیندێکسی 11 بەدواوە دەستپێدەکات

output

true
false
true

subsequence() میسۆدی

ئەم میسۆدە بەکاردێ بۆ دەرهێنانی کاراکتەرکانی نرخی سترینگێك لەنێوان

ماوەیەکدا کەخۆمان ماوەکە دیاری دەکەین.

یاسای درووستکردنی ⟵ Char subSquence(int startindex,int endindex)

```
class mu {
Public static void main(String[]args) {
String str =”welcome to kalar”;
System.out.println(str.subSquence(0,10));
System.out.println(str.startsWith(3,8));
}}
```

ئەگەر ماوەکە لەگەڵ نرخی سترینگەکە نەگونجا ئەنجام −1

output

welcome to

Come to

toCharArray() میسۆدی

ئەم میسۆدە بەکاردێ بۆ گۆڕینی نرخی سترینگ بۆکاراکتەر پیشاندانی یەك لەدوای یەك

یاسای درووستکردنی ⟵ Char[]toCharArray()

```
class mu {
public static void main(String[]args) {
String str =”Welcome to this  tutorial”;
System.out.println(str.toCharArray());
}}
```

output

Welcome to this tutorial

میسۆدی toLowerCase()

ئەم میسۆدە بەکاردێ بۆ ئەوەی هەموو نرخی سترینگەکە بکاتە پیتی بچووك.

```
class mu {
public static void main(String[]args) {
String  str =”KAlar”;
System.out.println(str . toLowerCase() );
}}
```

output

kalar

میسۆدی toUpperCase()

ئەم مسۆدە بەکاردێ بۆ ئەوەهەموو نرخی سترینگەکە بکاتە پیتی گەورە.

```
class  mu  {
public static void main(String[]args) {
String  str  =”kalar”;
System.out.println(str .toUpperCase() );
}}
```

output

kalar

محمد فایەق

زانکۆی گەرمیان – زانستی کۆمپیوتەر

میسۆدی valueOf ()

ئەم میسۆدە بەکاردێ بۆ پیشاندانی نرخی ناو فاریێبڵەکان.

```
class  mu {
public static void main(String []args) {
Strind  str =”kalar”;
int  h =15;
char [] ch ={‘w’,’r’,’y’,’a’};
double d =22.5;
boolean  b =true;
System.out.println(str);
System.out.println(h);
System.out.println(ch);
System.out.println(b);
    } }
```

output

kalar
15
wrya
true

تێبینی// ئەم سترینگانەی کەباسمان کرد زۆر گرنگن بۆیەگرنگی زۆری پێبدە.

ئەرەی Array

ئەرەی لەجافادا پێکدیت لە کۆمەڵێ داتا و زانیاری کە هەمان داتا تایپیان هەیە.

Array هەمووکات لە ئیندێکسی 0 دەستپێدەکات.

سوودی Array

1– Array بەکاردێ بۆخەزنکردن و ڕێکخستنی داتا و زانیاری.

2- دەتوانین ڕاستەوخۆ کام داتامان بوێت دەریبینین.

3– Array زۆر خێرایە لەجێبەجێ کردنی ئیشەکانی دا.

بێ سوودی Array

Array تەنها یەك بێسوودی هەیە ئەویش ئەوەیە ئەو زانیاریانەی تێدا خەزن دەکرێت دەبێ هەموویان یەك data type یان هەبێ.

جۆرەکانی Array

بەشێوەیەکی گشتی دوو جۆر Array مان هەیە:

1-single Dimensional Array

2- Multidimensional Array

Single Dimensional Array

ئەمە جۆری یەکەمی Array کەتەنها یەك [ ] هەیە کەبەکاردێ بۆخەزنکردنی داتا و زانیاری بەشێوەی ستونی یان بەشێوەی ئاسۆیی.

شێوازی ڕاگەیاندنی Declaration

بەدوو شێواز دەتوانین ڕایبگەیەنین:

1- تەنها ڕایدەگەیەنین و قەبارەکەی دیاریدەکەین دواتر نرخی پێدەدرێ کە پێیدەوترێ instantiation.

```
Data Type [ ] name =new  Data Type[array size];

Name [0]=data1;

Name [1]=data2

Name [2]=data3;
```

2- ڕادەگەیەنرێت وڕاستوخۆنرخی پێدەدرێت پێیدەوترێ Initialization.

```
Data Type [ ]  name ={data0,data1,data2,data3,....};
```

تێبینی// لەوانیە حاڵەتی default value ڕووبدات لە شێوازی ڕاگەیاندنی یەکەمدا , کاتێك ڕوودەدات کە ئێمە قەبارەی Array کەمان 3 بێت و تەنها دوو data مان هەبێ ئەگەر ڕوویدا ئەوا ئەنجامی داتای سێیەم null دەبێ.

Multidimensional Array

ئەمەیان جۆری دووەمی Array یە کە دوو [] []هەیەکە , بەکاردێ بۆ ڕێکخست و خەزنکردنی داتا و زانیاری بەشێوەی ستونی و ئاسۆی.

شێوازی ڕاگەیاندنی

ئەم جۆرە Array یەش بەهەمان شێوە بەدوو شێواز ڕادەگەیەنرێت:

1-تەنها ڕایدەگەیەنین و قەبارەکەی دیاریدەکەین دواتر نرخی پێدەدرێ کەپێیدەوترێ instantiation.

```
Data type [][] name =new Data type[array size row][array size column]

Name [0][0]=data0;
Name[0][1]=data1;
Name[0][2]=data2;
```

2-ڕادەگەیەنرێت وڕاستوخۆنرخی پێدەدرێت پێیدەوترێ Initialization.

```
Data type[] [] name ={ {data0,data1,...},{data0,data1,..} };
```

ئیش کردن لەسەر Array

زیاتر دەتوانین بە دوو ڕێگا ئیش بکەین لەسەر Array

1-بەکارهێنانی for loop

2-بەکارهێنانی forache (enhanced loop)

```
class  mu {
Public static  void main(String[]args) {
int [] arr ={1,2,3,4,5,6,6,7};
for (int i =0;i<=arr.lenght-1;i++) {
 System.out.print (arr[i]);
      } } }
```

output

12345667

```
for (declaration : expression) {
        //statements
          }
```

← یاسای دروستکردنی enhanced loop

```
class  mu  {
public static void main(String [] args) {
int  [] arr ={2,4,5,7,4};
for (int  result  :  arr) {
System.out.print(result);
  } } }
```

output

24574

بۆ نموونەی زیاتر و شێوزەکانی دیکە ئەتوانی خۆت بەدوایدا گەڕان بکەیت.

کۆنستراکتەر Constructor

کۆنستراکتەرمیسۆدێکی تایبەتەبەکاردیَ بۆپێدانی نرخ بەinstance variable و static variable, کاتێك حاڵەتی default value ڕوودەدات.

حاڵەتی default value ⟵ int d;

کاتێك میسۆدی کۆنستراکتەر درووستدەکریَ کە object درووستکرابێت چونکە object بەکاردێنین بو بانگکردن و پیێدانی نرخ بە Variable.

جۆرەکانی Constructor

1- Default constructor واتە پارامیتەری نییە.

2- parameterized constructor واتە پارامیتەری هەیە.

مەرجی دروستکردنی Constructor

بۆدروستکردنی Constructor دوومەرج مان هەیە:

1- دەبێت ناوی میسۆدی کۆنسترەکتەرەمان ناوی کلاسەکەی هەبێت.

2- نابیَ میسۆدەکەمان return type هەبیَ.

کەی کۆنستراکتەر بەکاردی

کاتێك حاڵەتی default value ڕوودەدات واتە کاتێك فارێبڵی static وinstance بێ داتان بەلاَم ئێمە بەکارمان هێناوە لەوکاتەدا ئەم حاڵتە ڕوودەدات بۆیە دەتوانین کۆنستراکتەر بەکاربێنین بۆلابردنی ئەم حاڵتە و ناردنی داتا بۆیان.

```
  class  mu  {
Static  int  id;
String  name ;
void  display () {
System .out.println(id+name) {
       }
Public static void main(String []args) {
mu  obj  =  new  mu();
obj .  display();
        }}
```

Default value → Output: 0 null

لەم نموونەدا فارێبڵی static م بەکارهێناوە وەك وەلاَمێك بۆ ئەوانەی دەڵێن کۆنستراکتەر تەنها بۆ instance variable بەکاردێ .

Default Constructor

جۆرێکی کۆنستراکتەرە بەکاردێ بۆ لابردنی حاڵەتی default value بەبێ بەکارهێنانی پارامیتەر بەڵکو ڕاستەوخۆ نرخەکەی پێدەدرێت .

```
Class  mu  {
Static int id ;
String  name ;
Mu () {
id=2;
String = "ahmad";
       }
void display() {
Systame .out.println(id+name);
          }
Public static void main(String [] args) {
mu  obj = new mu ();
    obj . display();
      } }
```

بەپێدانی نرخ حاڵەتەکەمان لابرد

output

2 ahmad

تێبینی // هەمووکات jvm ئەم جۆرە کۆنستراکتەرە دروستدەکات.

Parameterized Constructor

جۆرێکی کۆنستراکتەرە بۆلابردنی حاڵەتی default value بەکاردێت , بە بەکارهێنانی پارامیتەر و ناردنی نرخەکە لەڕێگەی دروستکردنی object وە.

```
Name of class (parameter list){
}
```

یاسای دروستکردنی

```
class  mu  {
int  id ;
String  name ;
mu (int i ,String r) {
id = i;
name = r;
}
Void display( ) {
System.ou.println(id+name);
}
Public static void main(String []args) {
mu  obj = new  mu (4,"karwan");
obj  . display();
} }
```

output

4 karwan

Overloading

بەکارهێنانی زیاد لەیەك میسوۆد لە پرۆگرامێکدا پێیدەوترێ Overloading

بەلاّم دەبێ ئەو سێ مەرجەی خوارەوە جێبەجێ بێت لەمیسوۆدەکەدا.

مەرجی Overloading

1- ژمارەی parameter list جیاوازبێ.

Sum (int c,String h,int a){} ⟸ سێ parameter list

Sum (int c,String h){} ⟸ دوو parameter list

2–data type کەیان جیاوازبێت.

Sum (int c,String h){}

Sum (int g,String a){}

3-ناوی میسوۆدەکان هەمان ناو بن.

Sum (parameter list){}

Sum (parameter list){}

تێبینی //نابێ یەك فارێبل دوو data type جییوازی هەبێ.

Sum (int a, String a){} ⟸ نابێ a دووdata type جیاوازی هەبێ

نموونە لەسەر یەکەم خاڵ

```
class  mu  {
Void sum (int a,intb,int c){
System.out.println(a+b+c);
}
Sum (int a,int b){
System.out.println(a+b);
}
Public static void main(String[]args) {
mu  obj = new mu();
obj .sum(10,10,10);
obj . sum (20,20);
}}
```

output

30

40

تێبینی//ئەگەر یەك نرخمان هەبوو واباشترە میسۆدەکان یەك جۆر treurn type یان هەبێ چونکە دواتر error درووست دەبێ.

```
int sum(int a,int b){}
double sum(double a ,double b){}
obj .sum(20,12);//compile error
```

نموونە لەسەر خاڵی دووەم

```
class mu {
void result (int a ,String  b) {
Syste.out.println(a+b);
}
void result (int b ,String a) {
System.out.println(b+a);
}
Public static void main(String[]args) {
mu  obj = new mu ();
obj . result(12,"ahamad");
obj . result(15,"mhamad");
} }
```

output

12 ahmad
15 mhamad

تێبینی //سوودی overloading ئەوەیە بەکارهێنانی پرۆگرامەکەمان زیاتر دەبێ.

Overloading main() method

بەهەمان شێوە دەتوانین کرداری Overloading بەسەر میسۆدی سەرەکیدا ئەنجام بدەین.

```
class mu {
public static void main(int a){
System.ot.println(a);   }
public static void main(String[]args) {
main(15);           } }
```

output

10

This keyword

کیۆردی this کۆمەڵێ تایبەتمەندی وسوودی هەیە لەخوارەوە باسیان دەکەین:

1- بەشیوەیەکی گشتی ئەم کیۆردی this بەکاردێ بۆ جیاکرنەوەی instance variable و local variable کاتیك هەردووکیان هەمان ناو و هەمان data type یان هەیە کاتێك نرخ دەنێرین جافا نازانێت نرخەکە بدات بەکامیان بۆیە دەتوانین ئەم کیۆردە بەکاربێنین بۆ پێدانی نرخەکە بە instance variable ئەگەرئەم کیۆردە بەکارنەهێنین ئەوا default valueڕوودەدات.

```
class  student {
int  id ;
String  name;
void  student (int id,String name) {
this .id=id;
this . name =name;
}
void display  (){
System.out.println(id+String);
}
public static void main(String [] args) {
Student  obj = new Student (4,”kalar”);
obj . display();
}}
```

output

4 kalar

2- کیۆردی This بۆبانگکردنی Constructor لەناو کلاّس دا.

```
class  Student {
int  id ;
String  name ;
Student () {
Syste.out.println("calling constructor") ; }
Student (int  id ,String  name ) {
this();
this . id= id;
this . name =name;
}
void display () {
System.out.println(id+name);
}
public static void main (String[]args) {
Student  obj  = new Student(12,"karwan");
obj . display();
}}
```

ئەم میسۆدە بانگ دەکەین ← Student () {

ئەم this بەکاردێ بۆ بانگکردنەکە ← this();

output

calling constructor
12 karwan

تێبینی // پێویستە هەمووکات this() لەپێشدا بنوسرێت .

3– کیۆردی this دەتوانین بەکاری بێنین وەکو argument.

```
class   s {
void  m ( s obj) {
System.out.println("method call");
}
void p () {
m ( this ) ;
}
public static void main(String[]args)  {
s s1 = new s();
S1.p();
}}
```

output

method calling

4- دەتوانین بۆ بانگکردنی میسۆدێك بەکاری بهێنین بەشێوەیەکی ناڕاستەوخۆ.

```
class  s {
void m() {
System.out.println("method is calling");   }
void n() {
this.m();    }
void p() {
n();    }
public static void main(String[]args) {
s   mm =new s();
s.p();
}}
```

ئەم میسۆدە بانگکراوەبەناڕاستەوخۆی

output

method is calling

ئەم بابەتە دوو بابەتی گرنگ لە خۆدەگرێ تاڕادەیەك باسكرنی بەنووسین قورسە بەڵام هەوڵدەدەین ڕوونی بكەینەوە.

Call By Value

Call by value واتە ئەو میسۆدەی كە بانگكراوەو نرخی پێدراوە , لەم بابەتە دا پرۆگرام ئەو كارە ئەنجام دەدات كە ئێمە دەمانەوێ ئەنجامی بدات وە هەر گۆڕانكاریەك لە پرۆگرامەكە ڕوویدات كاریگەری لەسەر پرۆگرام و ئەنجامی پرۆگرام نابێت .

```
class  oop {
public void callby value(int x){
  X=100;
      }

Public static void main (String [] args){
 X=50;
 oop  t = new oop();
 t.callby value(x);
System.out.println(x);
               }}
```

ئەم نرخەمان بۆدەرناچێ چونكە ئیمە داوای ئەم نرخەمانە نەكردووە

output

50

بەكورتی هەرگۆڕانكاریەك لەپرۆگرامەكەدا ڕوویدات جافا تەنها ئەوەمان بۆدەكات كە داوامان كردووە.

Call By Refrence

پێچەوانەی بابەتەکەی پێشووتر لە Call by refrence ئەو نرخەی دەینێرین بۆ میسوۆدەکە گۆڕانکاری درووستدەکات لەناو میسوۆدەکە کەکاریگەری دەبێت لە سەر گۆڕانی ئەنجامی پرۆگرامەکە .

```
class  oop {
 int x= 10;
public  void callby refrence(oop  ts){
      ts.x=20;
            }
public stativ void main(String [] args){
 oop  ts = new opp();
System .out.println("Before"+ts . x);

ts .callby reference(ts);
 System .out.println("After"+ts . x);

        } }
```

output

Before 10
After 20

Inheritance

بابەتێکی هەرەگرنگە کە دەتوانین لەڕێگەی ئەم کردارەوە داتا و زانیاری لەنێوان کلاسەکاندا بگوازینەوە لەرێگەی بەکارهێنانی کیۆردی extends وە.

- ئەو کلاسەی کەداتاکان دەبەخشێت پێیدەوترێت super class یان parent class.
- ئەو کلاسەی کە داتاکان وەردەگرێت پێی دەوترێت sub class یان child class.
- Inheritance بەمانای (میرات) دێت واتەچۆن باوکێك(superclass) ماڵ و موڵکی دەبەخشێت بە منداڵەکانی(child class).

یاسای دروستکردنی

```
class super {
    .........
    ..........
}
```

Super class

```
class sub extends super {
    ..........
    ..........
}
```

Sub class

نموونەیەك لەسەر inheritance

```
class  super {
int  data =15;
}
```

Super class

```
class sub extends super {
public static void  main(String[]args) {
sub  obj = new sub();
System.out.println(obj . data);
}
}
```

Sub class

output

15

لەناو super class دا فرێبڵێکمان هەیە بەناوی data کە 15 تێدایە توانیمانە بەکرداری inheritance ئەم داتایەیە بنێرین بۆ sub class لەرێگەی درووستکرنی object لەناو sub class و بانگکردنی فارێبڵەکە.

سوودی Inheritance

1-بۆ ئەنجامدانی کرداری Overriding.

2– سوودی هەیە بۆ کاتی بەکارهێنانی polymorphism.

3–بەخشینی داتا و زانیاری و دووبارە بەکارهێنانەوەی لە کلاّسێکی تردا.

کیۆردی Super

کیۆردی Super کۆمەڵێ سوودی هەیە کەدەتوانین لە inheritance دا بەکاری بهێنین, ئەم کیۆردە تاڕادەیەك لە کیۆردی this دەچێت.

سوودی کیۆردی Super

1-کیۆردی supper بەکاردێ بۆ جیاکردنەوەی میسۆد و فاریێبڵەکانی ناو super class و sub class ئەگەر هەمان ناویان هەبێ.

2-دەتوانین کیۆردی super بەکاربێنین بۆ هێنانی instance variable لەناو super class وە بۆناو sub class.

3-دەتوانین کیۆردی super بەکاربێنین بۆهێنانی میسۆدێك لەناو super class وە بۆناو sub class.

4-دەتوانین کیۆردی super بەکاربێنین بۆهێنانی میسۆدی constractor لەناو super class وە بۆناو sub class.

1-کیۆردی supper بەکاردێ بۆ جیاکردنەوەی میسۆد و فارێبڵەکانی ناو super class و sub class ئەگەر هەمان ناویان هەبێ.

```
class super {
int data =10;
void display () {
System.out.println("super"); }
class sub extends super {
int data = 15;
void display() {
System.out.println("sub"); }
void result() {
sub obj = new sub();
obj .display();
super.display();
System.out.println(obj.data);
System.out.println(super.data);
}
Public static void main(String[]args){
sub obj =new sub();
obj . result();
}
}
```

Super class

Sub class

output

Sub

15

Super

10

2-دەتوانین کیوردی super بەکاربێنین بۆ هێنانی instance variable لەناو super class وە بۆناو sub class.

یاسای درووستکردنی ← System.out.println(super.variable);

```
class super {
int data =100;
}
class sub extends super{
void display(){
System.out.println(super.data);
}
public static void main(String[]args) {
sub  obj = new sub();
obj . display();
}
}
```

Super class

Sub class

output

100

3-دەتوانین کیوٚردی super بەکاربێنین بوٚهێنانی میسوٚدێك لەناو super class وە بوٚناو sub class.

یاسای درووستکردنی ← Super .method();

```
class super {
void massage() {
System.ou.println("super");     }
class sub extends super {
void display() {
super . massage();
          }
public static void main(String [] args) {
sub  obj = new sub();
obj .display();
        }
        }
```

Super class

Sub class

output

super

4-دەتوانین کیۆردی super بەکاربێنین بۆهێنانی میسۆدی constractor لەناو super class وە بۆناو sub class.

```
class super {
  super () {
System.out.println("method constructor") {
     }
 Class sub extends super {
  sub () {
  super() ;
     }
Public static void main(String []args) {
  Sub obj = new sub();
          }
       }
```

output

Method constructor

مەرجی constructor هەرچەندە پێشتریش باسم کردووە:

1–هەمان ناوی کلاسەکەی هەبێت.

2–نابێ return type هەبێت.

Types of Inhritance

بەشێوەیەکی گشتی پێنج جۆر Inheritance مان هەیە :

1-Single Inheritance

2-Multi Level Inheritance

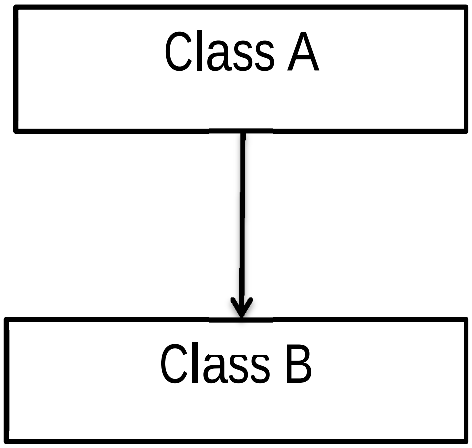
3-Hierarchical Inheritance

4-Hybrid Inheritance

5-Multiple Inheritance

Single Inheritance

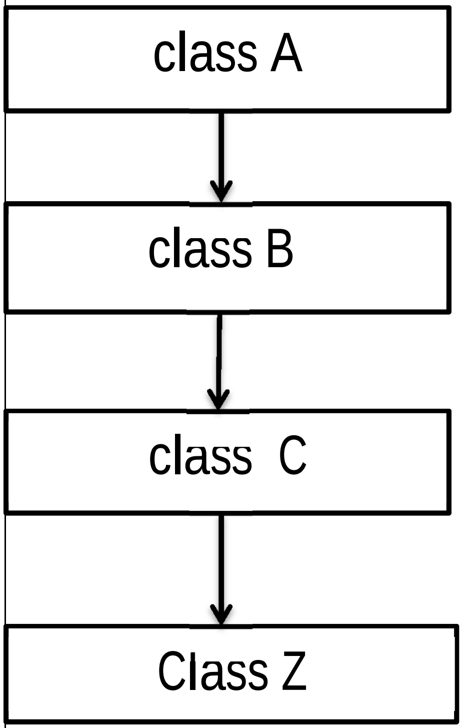
لەم جۆرەی Inheritance دا کلاَسێك دەتوانێت داتا و زانیاری بدات بە کلاَسێکی دیکە.




```
Class A {
    ...........
}
Class B extends A {
    ...........
}
```

Multi Level Inheritance

لەم جۆرە Inheritanceدا کلاسێك داتا دەدات بە کلاسێك ئەو کلاسەش دەیدات بە کلاسێکی تر بەشێوەی level بە level .



class A {............}

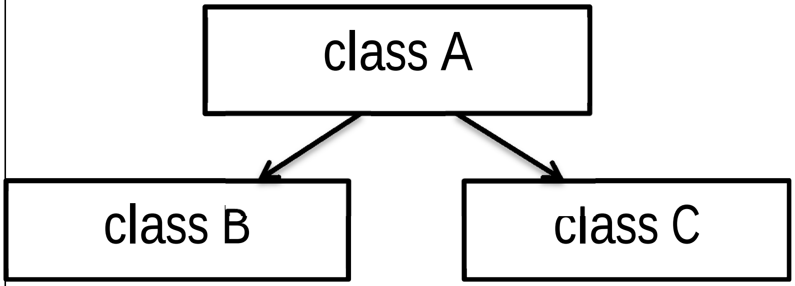
class B extends A {........}

class C extends B {.......}

class Z extends C {........}

Hierarchical Inheritance

لەم جۆرە Inheritance دا کلاسێك داتا دەدات بە دووکلاسی جیاواز.



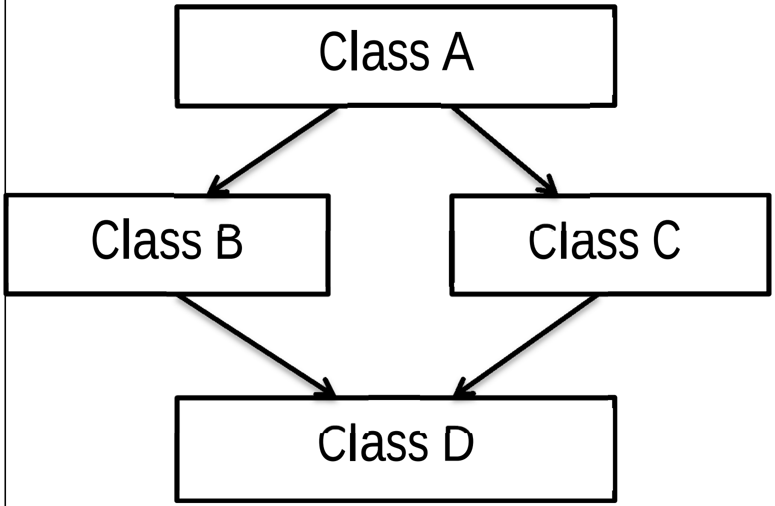
class A {........}

class B extends A {......}

class C extends A {.....}

Hybrid Inheritance

لەم جۆرە Inheritance دا کلاسێك داتا داتا دەدات بە دوو کلاسی دیکە ئەوانیش دەتوانن بیدەن بو کلاسێکی دیکە.



class A {......}

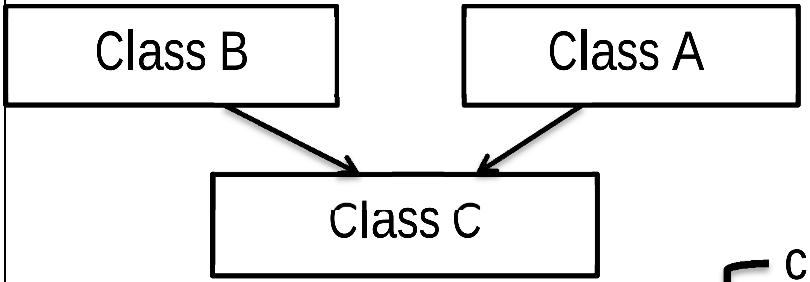
class B extends A {......}

class C extends A {......}

class D extends B,A{.....}

Multiple Inheritance

جۆرێکی دیکەی Inheritanceە کەکلاسێك لەیەك کاتدا دەتوانێ داتا لە دوو کلاسدا وەربگرێ بەلام ئەم جۆرەیانە ناتوانین لەجافادا بەکاری بهێنین چونکە لە جافادا کلاس ناتوانێ داتا لە دوو کلاسی جیاواز وەربگرێ بەلام دەتوانین لە interface دا ئەم کێشەیە چارەسەربکەین.



ناتوانین ئەم جۆرە کارەبکەین لەناو جافا

class A {......}

class B {......}

class C extends A,B {......}

Overriding

Overriding میسۆدێکە کە هەمان ناو هەمان return type وە هەمان parameter listی هەیە دەکەونە ناو super class و sub class بەلاّم بەوەی ناو sub class دەوترێت overriding.

سوودی method overriding

1-ئەم میسۆدە بەکاردیٚ بۆ ئەنجامدانی کارێك لەناو sub class دا کەلەلایەن super class ڕێگە بەم کارە دەدرێت.

2-لەکاتی بەکارهێنانی polymorphism دا دەتوانین ئەم میسۆدە بەکاربهێنین.

مەرجی بەکارهێنانی Overriding

1-دەبیٚ هەردوو میسۆدەکە هەمان ناو وە هەمان return type و وە هەمان parameter list هەبیٚ ئەگەر parameterی هەبوو .

2- نابیٚ میسۆدەکان کیۆردی staticیان لەگەڵ بێت چونکە ئێمە پێویستمان بە درووستکرنی object هەیە وەهەروەها نابیٚ کیۆردیfinal ی لەگەڵ بێت.

3-ئەگەر میسۆدەکە نەمان توانی کرداری Inheritance بەسەردابێنین ئەوا ناتونین کرداری Overriding یش بکەین چونکە کرداری Overriding لەکاتی کرداری Inheritance دا دەکرێت.

4-میسۆدی Constructor ناتوانیٚ کرداری Overriding ئەنجام بدات.

5–Overriding لەناودووکلاَسی جیاواز ڕوودەدات.

```
class  super {
void student () {
System.out.println("zankoy koya");
} }
class sub extends super {
void student () {
System.out.println("zankoy garmyan");
}
public static void main(String []args) {
sub  obj = new sub () ;
obj . run ( );
} }
```

output

Zankoy garmyan

لەم نموونەدا دوومیسۆدمان هەیەی بەناوی student کەهەمان ناو وە هەمان return type یان هەیە هەریەکەو لەناو super class و sub class دایە ئەگەر سەیری output بکەین نرخی ناو میسۆدی sub class کەی پیشانداوە, کەواتە ئێمەکرداری Overriding مان ئەنجام داوە.

تێبینی //ناتوانین کرداری Overriding بەسەر main() method دا بێنین چونکە کیۆردی static ی لەگەڵ .

جیاوازی نێوان Overloading لەگەڵ Overriding

Overloading

- Overloading توانایی بەکارهینانی پرۆگرامەکە زیاتر دەکات واتە : دەتوانین پرۆگرامەکە زیاتر بەکاربهێنین.
- لە Overloading دا دەبێ parameter ەکان جیاوازبن.
- کرداری Overloading لەناو یەك کلاّس دا جێبەدێ دەکرێت.

Overriding

- میسۆدی Overriding بەکاردێ بۆ ئەنجامدانی کارێك لەناو sub class دا کەلەلایەن super class ەوە ڕێگەی پێدەدرێت.
- کرداری Overriding لەنێوان دوو کلاّس دا ئەنجام دەدرێت لە کرداری Inheritance.
- لەکرداری Overriding دا دەبێ parameter ی هەردوو میسۆدەکە وەکو یەك بن ئەگەرparameterی هەبوو.

Abstraction

Abstractoin واتە بەکارهێنانی کیوۆردیAbstract کەلەگەڵ میسوۆد و کلاٚسەکاندا بەکاردیٚ, کاتیٚ ئەم کیوۆردە دەڕواتە سەر میسوۆد و کلاٚسەکان و interface شێوازی ئیشکردنی ئەو میسوۆد وinterface وکلاٚسەکان دەشارێتەوە و تەنها بە بەکارهێنەر شێوازی ئیشکردنەکە پیشاندەدات وەکو ناردنی نامە.

Abstract class

Abstractb class واتە ئەو کلاٚسەی کە کیوۆردیAbstract بەکارهێناوە کە کۆمەڵیٚ تایبەتمەندی هەیە:

- کلاٚسی Abstract لەوانەیە پێکهاتبیٚ لە میسوۆدێك یان چەند میسوۆدێك بەلاٚم بەبیٚ براکێت{ }ئەگەر کیوۆردی Abstract ی لەگەڵ بوو .
- مەرج نیە کلاٚسی Abstract میسوۆدەکانیشی Abstract بیٚ بەلاٚم ئەگەر میسوۆدەکەمان Abstract بوو دەبیٚ کلاٚسەکەمانیش Abstract بیٚ.
- کاتیٚ کیوۆردی Abstract دەڕواتە سەر کلاٚس وادەکات کرداریInheritance لەگەڵ کلاٚسێکی دیکەدا ئەنجام بدات وەهەموو میسوۆدەکان لەناو ئەو کلاٚسەدا بەکاردیٚ واتە لەناو sub class دا میسوۆدەکان بەکاردیٚ.

Abstract Method

Abstract method واتە ئەو میسۆدەی کە کیۆردی Abstract ی لەگەڵە

ئەم کیۆردە پێش ناوی میسۆدەکە دەنووسرێ کەکۆمەڵێ تایبەتمەندی هەیە.

- میسۆدی Abstract برکێتی{} نیەتەنها(;)لەگەڵە کاتێ لەناو super class دا دروستدەکرێت کەواتە ناتوانین بەکاری بهێنین لەناوsuper class دا بەلأم لەناو sub class دا براکێتی{} هەیە کەواتە دەتوانین لەناو sub class دا بەکاری

```
abstract void run() ;
```

⬅ لەناو super class دا

```
void run( ) {
        }
```

⬅ لەناو sub class دا

- ئەگەر میسۆدەکەمان کیۆردی Abstract لەگەڵ بوو دەبێ کلأسەکەمانیش کیۆردی Abstract لەگەڵ بێ.
- ئەگەر کلأسیك کیۆردی Abstract لەگەڵ بوو وەچەند میسۆدێکی هەبوو مەرج نیە هەموویان کیۆردی Abstract یان لەگەڵ بێت لەوانیە تەنها یەك دانەیان کیۆردەکەی لەگەڵ بێت.
- کاتێك کیۆردی Abstract دەڕواتە سەرکلأس ناتوانین ئوبجیکت لەجۆری Instantiation دروستبکەین بۆ پێدانی نرخ بەمیسۆدەکەمان.

```
sub  obj = new sub(“kalar,”hawler”);
```

✖

لە دوو شوێندا کیۆردی Abstract بەکاردێ :

1- Abstraction class(0 to 100%)

2- Interface(100%)

```
abstract class super {
abstract void  run ();
class sub extends super {
void run() {
System.out.println("zankoy garmyan");
        }
public static void main(String[]args) {
 super  obj = new sub();
 obj . run();
     } }
```

کلاسی abstract

میسۆدی Abstract بەبێ {}

Output

Zankoy garmyan

Encapsulation

Encapsulation بابەتێکی هەرەگرنگیOOP Concepts ە بەمانای

شاردنەوەی داتا و زانیاری دێت واتە ئەو داتا و زانیاریانەی کەشاردراوەتەوە و

تایبەتن ئێمە لەڕێگەی Encapsulation ەوە بتوانین دەست کاریان بکەین

و بیگۆڕین.

- دەبێ فارێبلەکەمان کیۆردیprivate بەکارهێنابێ واتەداتای ناو فارێبلکەمان ناتوانین دەستکاری بکەین و بیگۆڕن .
- ئەو فارێبلەی کە داتاکەی ناوی تایبەتە و ناتوانین بیگۆڕن و لەکلاسێکی تردا بەکاریبهێنین ئێمە لە ڕێگەی دروستکردنی میسۆدی setter و میسۆدی getter وە دەتوانین دەستکاری ئەو داتاوزانیاریانە بکەین کە تایبەتکراوە بتوانین لەکلاسێکی تردا بەکاریبهێنین .
- میسۆدی setter واباشترە void بێ وە میسۆدی getter واباشترە return type هەبێ چونکە لە getterدا دەمانەوێ نرخەکەمان پیشان بداتەوە وە دەبێ هەردووميسۆدەکە کیۆردی public یان

```java
public class bn {
    private int id =2;
    public void setter (int f){
        id = f;

    }

    public int getter (){
        return id ;

    }
}
```

دووكلاسی جیاوزە , بەفایلی bn.java خەزنی بكە

```java
public class oop {

    public static void main(String[] args) {
        bn s=new bn();

        s.setter(8);

int h=s.getter();

System.out.println(h);
    }
```

بەفایلی oopخەزنی بكە

output

8

سوودی Encapsulation ئەوەیە بتوانین داتا تیابەتكراوەكە پیشان بدەین و بیخوێنینەوە یان دەستكاری بكەین

Encapsulation دەسەلاتێك دەدات بە كلاسەكە كە چ جۆرە داتایەك خەزن دەكات.

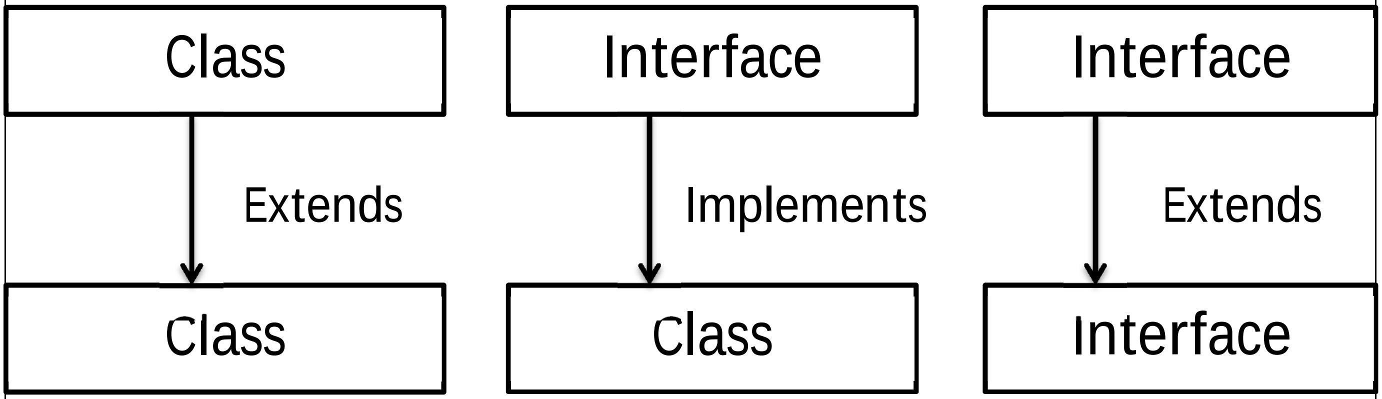
Interface

Interface بەمانای ڕووکەش دێت کە بنچینەی درووستکردنی کلاّسە , Interface پێکدێت لەکۆمەڵێ میسۆد و فاریێبڵ.

- میسۆدەکانی Interface براکێتیان { } نییە واتە Abstract و فارێبڵەکانیشی جێگیرن دواتر شێوازی ڕاگەیاندنی میسۆد و فارێبڵەکان باس دەکەین.
- Interface ڕێگەیەکە بۆ بەدەست هینانی Abstract بەرێژەی ٪100.
- Interface لەگەڵ کلاّسدا کۆمەڵێ خاڵی هاوبەش و خاڵی جیاوازیان هەیە .
- بەهەمان شێوەی Abstraction Class لە Interface دا ناتوانین ئۆبجێکت لەجۆری Instantiation دروستبکەین .
- Interfaceبوونی جۆرێك لە پەیوەندی دەنوێنێت لەنێوان خۆی و کلاّس دا.

خاڵی هاوبەشی Interface لەگەڵ Class

- Interface وەکو کلاَس پێکدێت لەکۆمەڵێ میسۆد بەلاَم بەبێ براکێت{ }.
- فایلی Interface بەهەمان شێوەی فایلی کلاَس خەزندەکرێت .
- بایت کۆدی Interface لەناو فایلی کلاَس دایە .



خاڵی جیاوازی Interface لەگەڵ Class

- لە Interface دا ناتوانین ئۆبجێکت لەجۆری Instantiation بەکاربهێنین بەلاَم لە کلاَسدا دەتوانین .
- لە Interface دا ناتوانین میسۆدی constructor بەکاربێنین.
- لە Interface داهەموو میسۆدەکان Abstract ە.
- لە Interface دا ناتوانین Instance variable بەلاَم لەکلاَس دا دەتوانین بەکاربهێنین .
- ئینتەرفەیس Extends ناکرێ لەگەڵ کلاَس دا بەڵکو Implements دەکرێ.
- لەکلاَسدا دەتوانین هەموو access modifierکان بەکاربێنین بەلاَم لە Interface دا تەنها public بەکاردێ.

ڕاگەیاندنی Interface

بۆڕاگەیاندنی Interface دەبێ کیوۆردی interface بەکاردێنین.

```
Interface student {
//variable is static ,final
//method is abstract without brace {}
}
```

تێبینی// Interface بەشێوەیەکی نائاشکرا Abstract بەڵام پێویست ناکات کیوۆردی Abstract بەکاربهێنین .

بەهەمان شێوەی Interface دەبێت میسوۆد و فارێبڵیش ڕابگەیەنرێت :

1- ڕاگەیاندنی میسوۆدی Abstract

بۆڕاگەیاندنی پێویست بەکیوۆردی Abstract و {} ناکات لەناو super classداڕاستەوخۆ(;) دادەنێین بەڵام لەناو sub class ادا دەبێ کیوۆردی public وبراکێت بەکاربێنین.

ڕاگەیاندنی میسوۆد لەناو super class

```
void display() ;
```

دواتر میسوۆدەکە لەناو sub class

```
Public void display () {
}
```

2- ڕاگەیاندنی Variable

بۆ ڕاگەیاندنی فارێبڵ دەبێت کیوۆردی static وfinal بەکاربێنین لەناو super class

```
static final int a=10;
```

Implementing Interface

Implementing Interface واتە ئیشپێکردنی کلاس لەگەڵ Interface بە

بەکارهێنانی کیوردی implements هەمیشە ئەم کیوردە لەناو کلاسەکەدا

ڕادەگەیەنرێت .

```
interface oop {
    void run();

    static final int a=3;

    static final int b=6;
}
class q implements oop{
    public void run (){
        System.out.println(a+b);

    }

    public static void main(String[] args) {
        q h=new q();
        h.run();


    }
```

Abstract میسۆدی

Abstractفاریبڵی

فایلەکە بە q خەزن دەکرێ

output

9

تێبینی // کلاسێك دەتوانێت لەیەك کاتدا Implementsبکات لەگەڵ چەندین

Interface دا بکات بەلاَم دەبێت پێشتر Interfaceکان ڕاگەیەندرابێت.

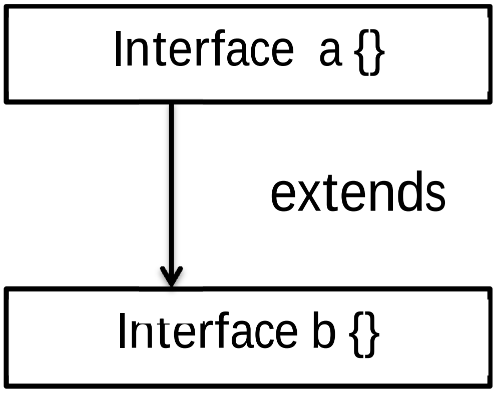
```
class  sub implements  a,b,c{

                    }
```

Extending Interface

کاتێك Interface لەگەڵ Interfaceدیکەدا کرداری Inheritance ئەنجام دەدات یان بابڵێین Interface داتا و زانیاری بدات بە Interface دیکە بە بەکارهێنانی کیوۆردیextends پێیدەوترێت Interface Extending.

```
public interface mu {
void  show () ;
   }
interface oop extends mu {
void show() ;
  }
```

ئینتەرفەیسی mu کرداریextending کردووە لەگەڵ ئینتەرفەیسی oop



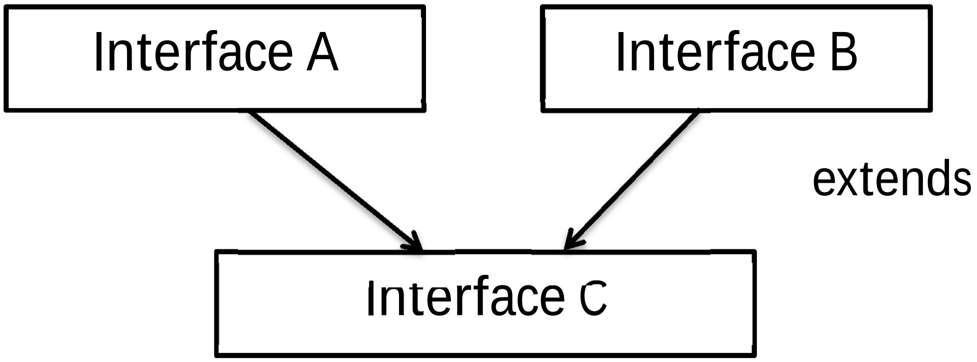
ئەگەر بمانەوێ ئەنجام مان دەست بکەوێ لم پرۆگرامەوە دەبێت Implement بکرێت لەگەڵ کلاسێك دا.

Multiple Inheritance Interface

دوو جۆر Multiple Inheritance Interface مان هەیە :

1-Extends

لەم جۆرەدا ئینتەرفەیسێك دەتوانی کرداری Inheritance ئەنجام بدات لەگەڵ چەند ئینتەرفەیسێکی دیکەدا بە بەکارهێنانی کیوۆردی extends.



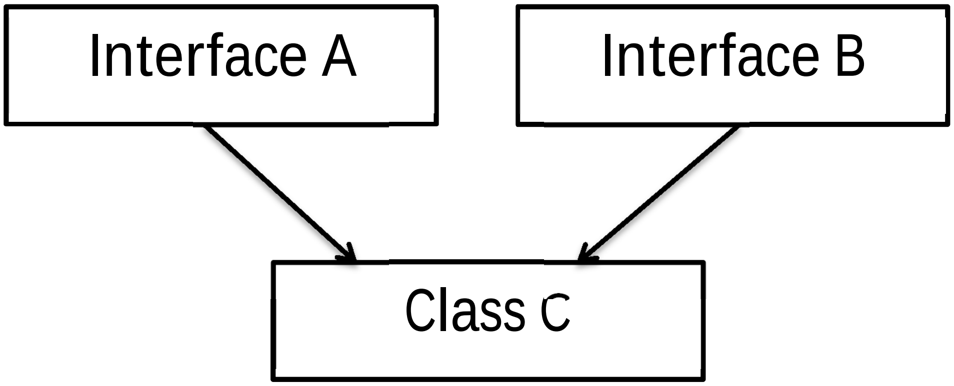

```
interface  super {
      ..............
   }
interface  sub extends super {
      ...............
   }
```

ئەگەر بمانەوی ئەم پرۆگرامە ئەنجامی هەبیّ دەبیّت Implements بکریّ لەگەڵ کلاّسێك دا.

2-Implements

لەم جۆرەدا کلاسێك دەتوانێت کرداری Inheritance ئەنجام بدات لەگەڵ چەند ئینتەرفەیسێك دا بە بەکارهێنانی کیوۆردی Implements.

تێبینی // کلاسێك ناتوانێ کرداری Inheritance ئەنجام بدات لەگەڵ چەند کلاسێکی دیکەدا بەلام لەگەڵ ئینتەرفەیس دا دەتوانێت.




```
interface  super {
    }
interface sub {
    }
class  A implements super ,sub {
    }
```

تێبینی //دەتوانین Interface لەناو Interface تردا دابنێین کەپێیدەوترێت

```
interface  mu {
interface  mm {
        }}
```

Nested Interface

Polymorphism

Polymorphismبەزمانی کوردی واتەچەند شێوەواتەلە Polymorphism دا میسوٚدێك دەتوانێت لەیەك کاتدا چەندین کرداری جیاواز ئەنجام بدات بەیارمەتی ئوٚبجێکت .

جوٚرەکانی Polymorphism

1-Compile Time Polymorphism (Dynamic Method)

Ex : Method Overloading

2-Run Time Polymorphism (Static Method)

Ex : Method Overriding

Compile Time Polymorphism

واتە میسۆدێك بەچەندین شێوەی جیاواز Compile Time Polemorphism
دەکرێ Compile بکرێت لەیەك کاتدا بۆنموونە وەکو Method Overloading.

```
class Overload
{
    void demo (int a)
    {
       System.out.println ("a: " + a);
    }
    void demo (int a, int b)
    {
       System.out.println ("a + b: " + a +  b);
    }
    double demo(double a) {
       System.out.println("double a: " + a);
       return a*a;
    }
}
class MethodOverloading
{
    public static void main (String args [])
    {
        Overload Obj = new Overload();
        double result;
        Obj .demo(10);
        Obj .demo(10, 20);
        result = Obj .demo(5.5);
        System.out.println("O/P : " + result);
    }
}
```

output

```
a: 10
a +b: 30
double a: 5.5
O/P : 30.2
```

لەنموونەی پەڕەی پێشووتردا میسۆدێکمان هەیە بەناوی demo کەمیسۆدێکی Overloading بەسیٚ شێوە دەرکەوتووە جارێك تەنها نرخی a پیشانداوە دواتر کرداری کۆکردنەوەی نێوان a و b ئەنجام داوە کرداری a لێکدان a ئەنجامداوە کەواتە میسۆدی demo کرداری Polymorphism ئەنجام داوە و بەسیٚ شێوە دەرکەوتوە.

مەرجی میسۆدی Overloading

بۆ دروست بوونی میسۆدی Overloading دەبیٚ میسۆدەکە چەند مەرجێك جێبەجیٚ بکات:

1- دەبیٚ ژمارەی Parameter list کان جیاوازبیٚ.

```
void  demo (int a,int b ){ }
void demo (int a, int b,int c){ }
```

2- دەبیٚ میسۆدەکان هەمان ناویان هەبیٚ .

```
void  demo () {}
void demo (int a) {}
```

3- دەبیٚ داتاتایپی parameter ەکان جیاوازبن .

```
void demo (int a){}
void demo (String a){}
```

Run Time Polymorphism

پرۆسەیەکە کەمیسۆدی Overridden بانگکراو سورە لەسەر ئەوەی run بکرێت لەجیاتی ئەوەی compile بکرێت.

Upcasting

Upcastingشێوازێکی دروستکردنی ئۆبجێکتە ئەم شێوازە بەتایبەتی لەکاتی کرداری Extends دا بەدیاردەکەوێ.

```
class  A {
 }
class B extends A {
 }
```

- A زیاتر وەکو پیشاندانی کلاسی A بەکاردێ
- obj لێرەدا دەوری فاریێبل دەبینێ کە داتاکانی ناو کلاسی B هەڵدەگرێت و پیشانی دەدات
- B ئاماژە بە کلاسی بانگراوەکە دەکات وە هەمیشە داتاکانی لەناو فاریێبلی obj هەڵدەگرێت
- بە A لەگەڵ obj دەوترێت Refrence variable.

```java
class Bike{

  void run(){

System.out.println("running");
}
}
class Splender extends Bike{

     void run(){

   System.out.println("running safely”)

     }
     public static void main(String args[]){
       Bike b = new Splender();//upcasting
       b.run();
     }
   }
```

output

Running safely

لەم نموونەیەدا میسۆدێکمان هەیە بەناوی run کە Overriding کراوەی لەناو دوو کڵاسدا دواتر ئۆبجێکتێکمان درووستکردووە بە شێوەی upcasting بانگی کڵاسێکمان کردووە کە extends کراوە وەهەمیشە داتای ناو ئەومیسۆدە پیشان دەدات کە کڵاسەکەی extends کراوە لە ئۆبجیکتەوە کڵاسەکە بانگ دەکەین .

تێبینی// لەpolymorphismئەگەر کیۆردیAbstract چووە سەرparent class کڵاس ئەوا ناتوانین ئۆبجێکت لەوکڵاسە دروستبکەین بەڵام دەتوانین لە child class دروستی بکەین.

package

Package لەجاڤادا پێکدێت لەکۆمەڵێ کلاّسی لەیەك چوو یان کۆمەڵێ

interface وsub package که ئەنجام دانی کارەکانمان ئاسان دەکات .

جۆرەکانی Package

1-Built-in Package

2-User –Defind Package

Built-in Package

لەناو زمانی جاڤادا کۆمەڵێ Package ئامادەکراو هەیە کەهەریەکەیان ئیشێکی

تایبەت دەکات که بەکارهێنەر دەتوانێت سوودی لێ وەربگرێت.

1-Java .lang

ئەم Package زیاتر لەکردارە بیرکارییەکاندا بەکاردی و پێکهاتوە لەکۆمەڵێ

(data type ,class , object) کەدەتوانین سودی لێ وەربگرین له

پرۆگرامەکەماندا

2-Java . util

Package ئامادەکراوی جافایە کە لە hash tableو String parsing

system properties دا کەدەتوانین سودی لیٚ وەربگرین

3-Java . net

ئەم Package دەتوانین لە بواری Networking دا بەکاریبهێنین .

4-Java . io

Input stearm

Output stearm

Package ئامادەکراوی جافایە کەدەتوانین بۆ داخل کردن و پیشاندانی کارێك

سوودی لێوەربگرین.

5- Java . awt

Package ئامادەکراوی جافایە کەدەتوانین لە GUI سوودی لێوەربگرین وەکو

دروستکردنی butoon , check box,combo box .

6-Java . applet

Package ئامادەکراوی جافایە کەدەتوانین لە دانانی وێنە ڤیدیۆ و دەنگ دا

سوودی لێوەربگرین.

تێبینی // هەمووکات لەپێش ئەم Package دا دەبێت کیوٚردی import

بەکاربێنین.

import java. util*;

User –Defind Package

لەم جۆرەی Package دا خۆمان دەتوانین Package درووستبکەین بەدوو ڕێگەبە بەکارهێنانی کیۆردیpackage وەدەبێ لەسەرەتای پرۆگرامەکەدا بنووسرێت :

1– درووستکردنی Package لەو text editor کە IDEA وەکو Netbeanse و Eclipes وIntelliJ .

لەم text editorادا پێویست ناکات خۆمان فایلی Package دروست بکەین ڕاستەوخۆ خۆی دروستی دەکات ئێمە تەنها ناوی Package دەنێین سەرەتا دەڕۆینە سەرnew دواتر Package هەڵدەبژێرین و ناوی دەنێین.

2– درووستکردنی Package لەو text editor نەی کە IDEA نین وەکو text pad و note pad .

لەم جۆرە text editor نەدا دەبێ خۆمان فایل دروست بکەین سەرتا فایلێکی ئاسایی دروست دەکەین دواتر فایلی کلاٌسەکەمان دەخەینەناوی کەلەناو پاکێج دایە دواتر کلاٌسێك درووست دەکەین کە میسوٌدی سەرەکی تێدایەو بانگی کلاٌسی ناو پاکێجەکە دەکەین لەڕێگەی دروستکردنی ئۆبجێکتەوە .

package nn ; — سەرەتا فایلێك درووست بکە بەناوی nn

public class ss { — دواتر فایلی بەرنامەکە کەناوی ss ە لەناو فایلی nn

```
public void display (){
System.out.println("hello java ");
}}
```

import nn.ss; — کلاسی ss لەناو پاکێجی nn بانگکردووەتە ناو کلاسەکەبە import

public class mu { — کلاسی mu خەزن بکە بەلاَم نابیَ لەناو فایلی nn دابنێین

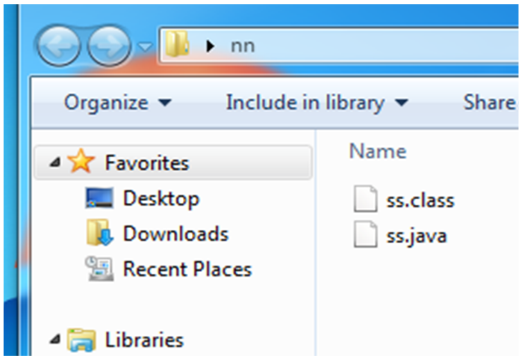
```
public static void main (String [] args) {
```

ss obj = new ss (); — لەناو میسۆدی سەرەکی کلاسی ss مان بانگ کردووە

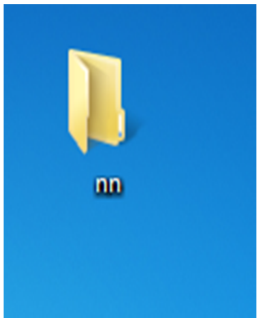
```
obj . display() ;
}}
```

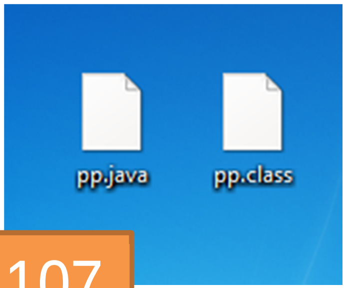
output

Hello wold



کلاسی ss لەناو فایلی nn
خەزن دەکەین





دواتر کلاسی pp خەزن دەکەین بەلاَم نابیَ لەناو
فایلی nn ادا دواتر compile و run دەکەین

کیۆردی Import

کیۆردی Import زیاتر وەکو بانگکردن یان داخل کردنی کلاّسێك لەناو پاکێجێك دا بۆ ناو کلاّسێکی دیکە و بەکارهێنانی وەکو نموونەی پەڕەی پێشوو .

تێبینی // پاکێجە ئامادەکراوەکانی جافا کە پێشتر باسمانکردن بەهەمان شێوە دەبێت کیۆردی Import بەکاربێنین بۆ بەکارهێنانی .

تێبینی // بەهەمان شێوەی کیۆردیImport دەتوانین access modifier کان بەکاربێنین دەتوانی سەیری بکەیت لەسەرەتادا باسیان کردووە.

Modifier	Class	Package	Subclass	World
public	✓	✓	✓	✓
protected	✓	✓	✓	✗
*no modifier**	✓	✓	✗	✗
private	✓	✗	✗	✗

به کارهێنانی دووکلاسی جیاواز لەناو دوو پاکێجی جیاواز دا

ئێمەدەتوانین دەتوانین کلاسێك لەناوپاکێجێك دابەکاربێنین لەناوکلاسی پاکێجێکی دیکەی جیاواز بەکورتیەکەی واتە بەکارهێنانی کلاسێك لەناو کلاسیکی تردا لەدوو پاکێجی جیاوازدا بەسێ ڕێگە دەتوانین ئەم کارەبکەین :

1-import package name . *;

2-import package name . class name ;

3-fully qualified name

---

import package name . *;

```
package oo;

public class pp {
    public void show () {
        System.out.println("hello");
        }}

package yy;
import oo.*;
public class mm {

    public static void main(String[] args) {
pp ob = new pp();
ob .show ();
    }}
```

دەتوانین * لەجیاتی ناوەی کلاسی pp بەکاربێنین بەکاریبێنین

output

hello

import package name . class name ;

Import كلاسێك دەكەین بۆناو كلاسێكی دیكە لەدوو پاكێجی جیاوازدا بەبەكارهێنانی

ناوی كلاسی import كراو.

تێبینی //بەكارهینانی ناوی كلاسی import كراو لەجیاتی * هیچ جیاوازیەكی نیە

,ئەستیرەكە (*) زیاتر بەشێوە گشتی بەكاردێ كاتێك نەمانەوێ ناوی كلاسەكە

```
package oo;

public class pp {

   public void show () {

      System.out.println("hello");

   }
 package yy;
 import oo.pp;
```

ناوی كلاسی pp مان بوكارهێناوە لەجیاتی *

```
 public class mm {

    public static void main(String[] args) {


    pp ob = new pp();
 ob .show ();
    }
 }
```

output

hello

fully qualified name

لەم جۆرەدا پێویست ناکات ناوی کلاسەکە import بکەین تەنها لەکاتی دروستکردنی ئۆبجێکتەکەدا بۆبانگکردنی کلاسەکە ناوی پاکێجی کلاسەکە دەنووسین.

```
package oo;

public class pp {
    public void msg(){
        System.out.println("kalar");
    }
                    }

package ii;

public class uu {

    public static void main(String[] args) {
        oo.pp ob =new oo.pp();
        ob.msg();

    }

}
```

oo ناوی ئەوپاکێجەیە کە کلاسی pp تێدایە

بەکارهێنانی کیۆردی Static لەگەڵ کیۆردی Import دا

لە پاکێج دا دەتوانین کیۆردی static لەگەڵ کیۆردی import دا بەکاربێنین

```
package oo;

import static java.lang.System.*;
class pp {
  public static void main(String args[]){

    out.println("Hello");// پێویست ناكات systemەكە بنوسین
    out.println("Java");

 }
   }
```

سوودی ئەم کیۆرد static ئەوەیە بۆ کەم کردنەوەی ئەو کۆدانەیە کە پێویستمانە ئەگەر میسوٚد و فارێبلێ پرۆگرامەکەمان static بوو.

تێبینی // بەکارهێنانی کیۆردی static و کیۆردی import پێکەوە وادەکات پرۆگرامەکەمان بێ کەڵك دەکات .

تێبینی//لێرەدا باسی سوودەکانی package ناکەین چونکە لەکاتی بەکارهێنان دا بەکارهێنەر دەزانێ سوودی چیە.

پێرست

پێشەکی..1
بەشەکانی OOP...2
کلاس...4
ئۆبجێکت...8
میسۆد...13
Modifier...19
ڕێساکانی ناونان Identifier...................................30
Variable...31
Character Class..35
String...41
Array..58
Constructor..62
Overloading...66
This keyword...69
Abstraction mechanisem.................................72
Inheritance...74

Oerriding...84
Abstraction...87
Encapsulation...90
Interface 92
Polymorphism...99
Package 104
پێڕست 113