



coupling ↓
cohesion ↑

changability

Inheritance

extend

field ویرایش هایت میروند

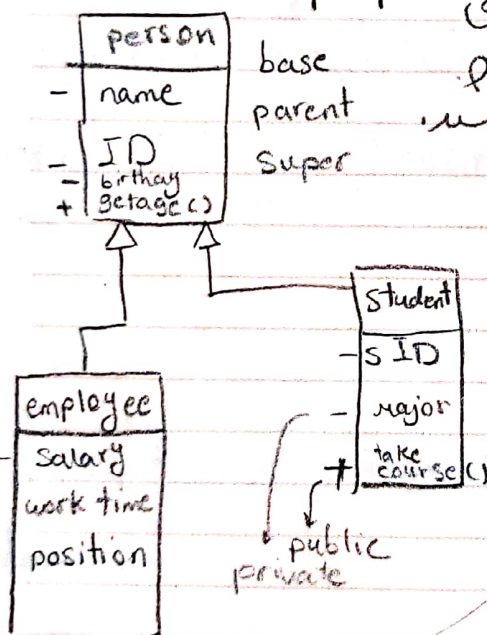
methods متدها (رشته ها) به ارث میروند (عصره ها) Attribute ها را تو سم (عصره) میروند



حق محدود کردن (کم کردن) ویرایش در رشته های

والد را نداریم و فرزندان تمام را باید داشته باشند.

فرزند حق استفاده مستقیم از private ندارد



Reusability ↑

Code Duplication ↓

در کلاس فرزند به کلاس والد دسترسی

explicit ← Super (n, niD, bd)

Drive/implicit ← Super()

child
روز بهزیستی و تأمین اجتماعی

subclass



(explicit)

البر صبور صبیح کاشته القدر والد را صدا میزنم، جاوا صبور صبیح (implicit) صدا میزند
کاشته القدر بدون پارسا والد را صدا میزنند. (البته البر کاشته القدر با برهان و مستقیم
چون کاشته القدر بدون اگر برهان پیغامش پس ما میباید اورو داریم)

در صورتی که تابع در کلاس والد نوشته باشیم
 این خصوصیات در کلاس فرزند نیز از این تابع استفاده کنیم، پس تابع
 to String را override می‌کنیم.
 (پس متدهای زیر کلاس، to String که در کلاس خود وجود دارد را valid
 می‌کنند و مبله را منسوخ می‌کنند.)
 اگر to String اولی private باشد، override نیست و خود
 to String تابع فرزند استفاده می‌شود.

person p1 = new person("Ali", 1, 90110)
 Reza

p1.getAge(); ✓

p1.getName(); ✓

Student s1 = new Student("Ahmad", 2, 2282)
 , 111, Computer)

s1.getAge(); ✓

s1.getName(); ✓

s1.getStudentNo(); ✓

```
public Student (name, NID, bd, SID, major) {  
    Super (name, NID, bd);  
    this.StudentNo = SID;  
    this.major = major; }  
    // دانشجو
```

Super. toString

تبدیل به String: خود را به صورت رشته تبدیل کنیم

Super. run

Super. run

طالع دسترسی دارد به نام protected
نفاذ داده می شود. (طالع دسترسی آن از private بیشتر و از
public کمتر است.)

```
public class person {
```

```
    private String name;  
    protected Date birthdate;  
    private String nationalID;
```

نام دسترسی عمومی

تاریخ تولد

```
    public person(String nam, String nationalID, Date birthdate) {  
        super();  
        this.name = name;  
        this.birthdate = birthdate;  
        this.nationalID = nationalID;  
    }
```

```
    public String getName() {
```

```
        return name;
```

```
    public String getNationalID() {
```

```
        return nationalID;
```

```
    public String toString() {
```

```
        return "Name = " + name + " , ID = " +
```

```
        nationalID;
```

public > protected > package > private

✓ هم جا

✓ درون بسته

X درون بسته

X درون بسته

✓ داخل بسته

✓ داخل بسته

X داخل بسته

X خارج بسته

X خارج بسته



بانک صادرات ایران

public override

protected

۱۳۹۶ تیر

۱۴۳۸ / شوال / ۲۵

20 / July / 2017



۲۹

پنجشنبه



public class main {

public static void main(String[] args) {

person p = new person("Ali", "123")

student s = new student("Amir", "999", "4", "c")

شهادت حضرت امام جعفر صادق علیه السلام (۱۴۸ هـ ق) (تعطیل)

۱۳۹۶ تیر

۱۴۳۸ / شوال / ۲۶

21 / July / 2017



۳۰

جمعه



بانک صادرات ایران



```
public class student extends person {
    private String studentNo;
    private String major;
```

```
    public student(String name, String nationalID,
        String studentNo, String major) {
```

```
        super(name, nationalID);
        this.studentNo = studentNo;
        this.major = major;
    }
```

```
    @Override
    public String toString() {
        return "studentNo = " + studentNo + "major"
            + major + ["name = " + getName() + "nationalID = "
                + getNationalID()];
    }
```

```
    return super.toString();
```



poly morphism عمل‌های ساده سازی تفاوت

✓ Person p = new Student (); Is A

X Student s = new person (); Is not A

p. dosomething → dosomething Student

person p2 = new employee ();

p2. dosomething → do something employee

ساده سازی‌ها متفاوت می‌شوند (بعضی از نزدیک به یک Reference اند) /
 به یک object در زمان Runtime، نام می‌آید انتخاب شود، انتخاب می‌شود.

(Dynamic binding)
 به ما اجازه می‌دهد که در زمان اجرا (Run time) انتخاب شود.

Compile time

(Static binding)

private, static متغیر

person pe = new Employee ();

Employee ep = pe ; X

Employee ep = (Employee) pe ; ✓

Type cast نوع انباشته شدن

poly morphism ← up casting

developer ← down casting





دوشنبه

۱۳۹۶ مرداد

۲۹ / شوال / ۱۴۳۸

24 / July / 2017

Employee es = (Employee)p2; X down cast

if (p2 instanceof Employee) {

Employee es = (Employee)p2;

System.out.println("Is instance of");}

else

System.out.println("Is not instance of");

→ check down casts

* تعیین: وراثت inheritance ، اصل ، طایفه ، اجداد ، مادر ، ملکه

Shapes
abstract
public class shape {

public double getArea();
return 0; }

public double getPerimeter();
return 0; }

public abstract double getArea();

همچنین میتوان از این روش استفاده کرد و فراموشی حواسه داشته باشد

public abstract double getPerimeter();



```
public class Circle extends Shape {
    private double radius;
    public Circle(double radius) {
        super();
        this.radius = radius;
    }
```

@Override

```
public double getArea() {
    return Math.PI * Math.pow(radius, 2);
}
```

سازمان عملیات انتخاب آفرین مرصاد (۱۳۹۷ هـ.ش)



@Override

```
public double getPerimeter() {
    return 2 * Math.PI * radius;
}
```

+ getter & setter & toString

روزگار آفرینی و آموزش های نو و حرفه ای



```
public class Rectangle extends Shape {
    private double width, length;
```

@Override

```
public double getArea() {
    return width * length;
}
```

@Override

```
public double getPerimeter() {
    return 2 * (width + length);
}
```

```
public Rectangle(double width, double length) {
    super();
```

```
    this.width = width;
```

```
    this.length = length;
}
```

+ getter & setter & toString



۱۳۹۶ مرداد

۱۳۳۸ / ۵ / ۲۹

29 / July / 2017

```
public class main {
```

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

```
        Rectangle r = new Rectangle(1, 2);
```

```
        Circle c = new Circle(3);
```

```
        System.out.println(r); // to String
```

```
        System.out.println(c);
```

```
        System.out.println(r.getLength()); // 2  
        Shape sh = r;
```

```
        System.out.println(sh);
```

```
        System.out.println(sh.getArea());  
                                getPerimeter();  
                                getLength();  
                                getWidth();
```

```
        ArrayList<Shape> shapes = new ArrayList<Shape>();
```

```
        shapes.add(r);
```

```
        shapes.add(c);
```

```
        shapes.add(new Rectangle(10, 20));
```

```
        if (abstract) shapes.add(new Shape());
```

روز تحلیل از امام زادگان و بقیع مشهوره
if abstract
شکل
حسابها از این فرم
دارد



۱۳۹۶ مرداد

۶ / ذی القعدة / ۱۴۳۸

30 / July / 2017



یکشنبه



for (shape s: shapes)

System.out.println(s.getArea());

System.out.println(s.getPerimeter());

روز بزرگداشت حضرت احمد بن موسی شاهچراغ علیه السلام

بانک صادرات ایران



* می‌توانید این‌ها را در Interface تعریف کنید: public, Abstract, static, final, data member

در این پروژه (در سطح Object نیست) data member را تعریف نمی‌کنیم. public, static, final تعریف می‌کنیم.

new project → Interface

```
public interface Drawable {
```

```
    public void draw();
```

~~System.out.println("this is rectangle");~~

```
    public static final colors = {red, blue, yellow};
```

* Interface می‌تواند قابل ترسیم باشد

```
public interface interface1 extends interface2 {
```

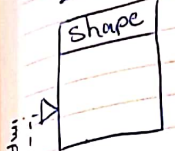


Interface (رابطه)

Interface پیش از Abstract هستند.

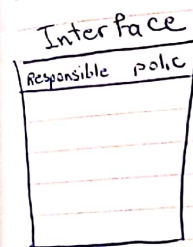
Interface می‌تواند دارای data member (field) باشد.

Interface می‌تواند دارای متد باشد و می‌تواند برای اشیاء تعریف شود. وابسته به نیاز است که چگونه استفاده شود.



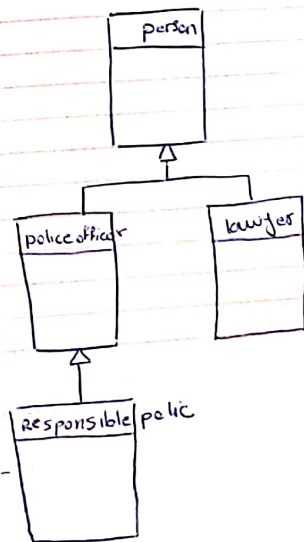
shape s = new Rect();

علاوه بر این، می‌تواند دارای data member باشد. فقط می‌تواند برای اشیاء تعریف شود و می‌تواند دارای متد باشد.



روزی اعضای خون

می‌تواند دارای data member باشد و می‌تواند دارای متد باشد.



مرداد
۱۰ / ذی القعدة / ۱۴۳۸
3 / August / 2017

۱۲
پنجشنبه

```
public class main {
```

```
    Rectangle r = new Rectangle();
```

```
    r.draw();
```

```
    Drawable d = new Rectangle();
```

```
    d.draw();
```

```
    Drawable d2 = new Circle();
```

```
    d2.draw();
```

مرداد ۱۳۹۶

۱۱ / ذی القعدة / ۱۴۳۸
4 / August / 2017

۱۳
جمعه

```
}
```

ولادت حضرت امام رضا عليه السلام (۱۴۸ هـ ق)

کتاب سادات ایران

مرداد ۱۳۹۶

۹ / ذی القعدة / ۱۴۳۸
2 / August / 2017

۱۱
چهارشنبه

```
public class Rectangle extends shape implements  
    Drawable {
```

```
    public void draw() {
```

```
        System.out.println("this is a Rectangle");  
        System.out.println("this is a Rect" + Colors[2]);
```

```
    public class Circle extends shape implements  
        Drawable {
```

```
        public void draw() {
```

```
            System.out.println("this is a circle");  
        }
```

کتاب سادات ایران

```
public class shape {  
    public double get Area() {  
        return 0; }  
}
```

* اساس از خاص ارث بری extend ۰-۱

* خاص رابط و پیاده سازی implement ۰-n

x رابط، رابط را توسعه می دهد extend

ساده java 8+

Concrete :

```
default void f() {
```

```
    =  
    =  
    }
```

معمول ترین از data mem
(public - static - final)
استفاده نمود



exception handling

دربرداشتن استثناها

```
public void setAge (int age) {
```

```
    if (age < max_Age)
```

```
        this.age = age;
```

```
    else
```

```
        this.age = max_Age;
```

```
}
```

توانی این را به جای استفاده از max_Age

```
int a , int b      a = 4      b = 0
```

```
return a/b;
```

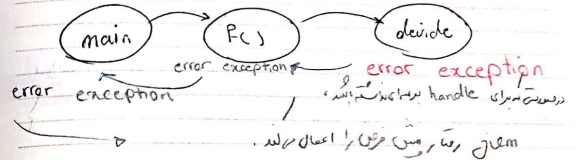
exception in thread java.lang.ArithmeticException

: / by zero



try {
f(); } catch (exception e) {
System.out.println("exception caught");
System.out.println(e.getMessage());
}
+ شماره متغیر یا نام exception / by zero
۳، ۴، ۵ از این try میتوان بین این catch پاست

```
static void f() {  
    devid(5, 0);  
}  
  
static double devid(int a, int b) {  
    return a/b;  
}
```



Stack Trace ← چک کنی که در مراحل screw شدن error exception

Try-catch (Block Try catch) آن مسخره که از مالد
اصولاً حالتی که دارد در این بلوک نداریم تا ریشه را در بیرون
آه جارا بیدار! غلط می‌نویس دانه سبز است.



۱۳۹۶ مرداد

۱۴ / ذی القعدة / ۱۴۳۸

9 / August / 2017

throws Stringlength exception

```
public void setName(String name) {
```

```
    if (name == null)
```

```
        throw new NullPointerException();
```

```
    if (name.length == 0)
```

```
        throw new StringlengthException();
```

خطا می‌دهد این اسم صحیح نیست و طول نام درست نیست. Exception باشد.

```
public void setAge(int age) {
```

```
    if (age < MaxAge)
```

```
        this.age = age;
```

```
    else
```

.. age is too large

```
        throw new InvalidAgeException();
```

throws exception
name

خطا می‌دهد این سن صحیح نیست و عدد سن صحیح نیست.

به قدری بزرگ است که صحیح نباشد.

```
* public class StringlengthException extends Exception {
```

```
}
```



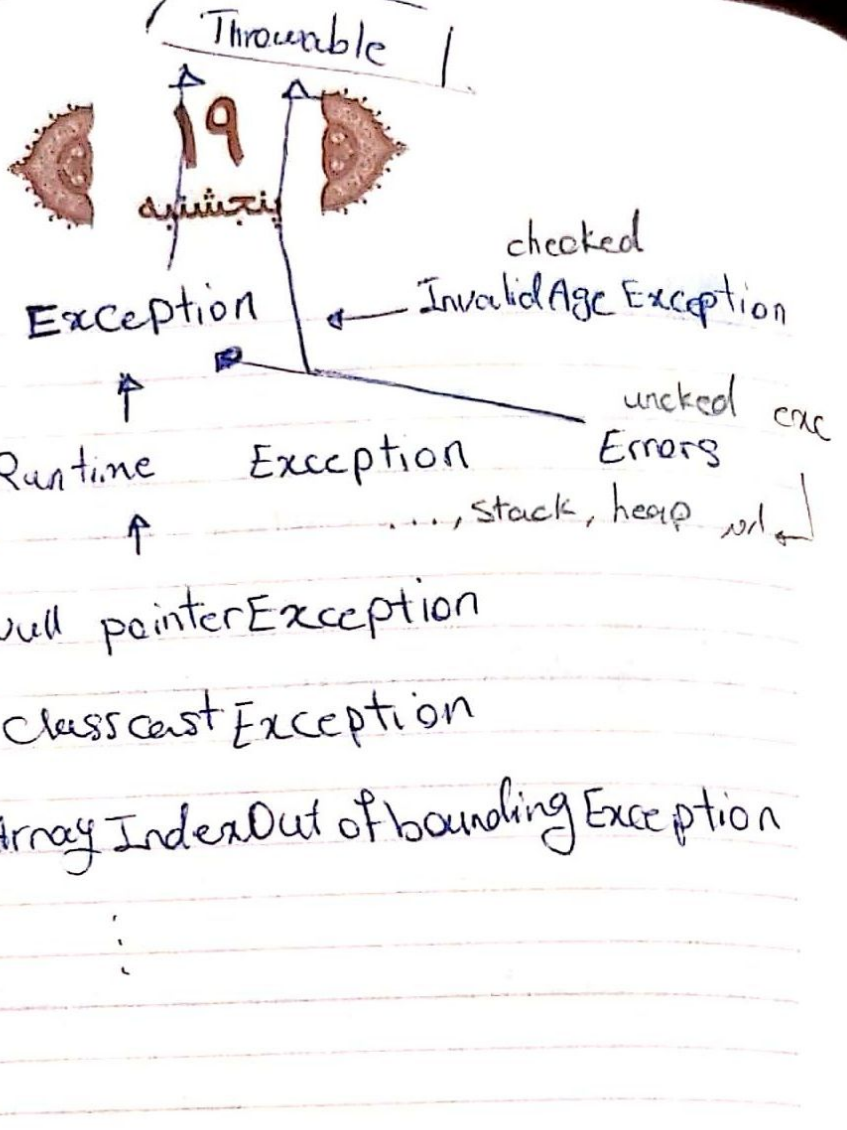
بنگاهت ایران

۱۳۹۶ مرداد

۱۷ / ذی القعدة / ۱۴۳۸

10 / August / 2017

checked
String length
exception



۱۳۹۶ مرداد

۱۸ / ذی القعدة / ۱۴۳۸

11 / August / 2017

۲۰ جمعه

Object oriented (analysis & design)

+ الفنايس + الوریسم های ریاضی - امتیاز

+ Design pattern



بانک صادرات ایران

~~Container~~

Container

array list

array list a = new array list ();

Array list < person > a = new Array list < person > ();

generic type < >

* جنس نامی ها object است و شیء تواند primitive باشد
و به این wrapper نام استفاده کنیم.



blocke Finally

برای انجام کارها در finally و در صورتی که در try

```
Static double divide(int a, int b) {
```

```
try {
```

```
double res = a/b;
```

```
return res;
```

```
}
```

```
catch (ArithmeticException e) {
```

```
System.out.println(2);
```

```
return 2; }
```

```
finally {
```

```
System.out.println(3);
```

```
return 3; }
```

ارائه Container :

data structure
List
ArrayList
LinkedList

Set (interface)
HashSet
TreeSet
equals
hashCode

Iterable
Collection
List
Set
ArrayList
LinkedList
HashSet
TreeSet

کلاس ها و متدهای Final :

override
final
override

extend
final
final

Static binding
private
Static
final

Dynamic binding — polymorphism

int P(Student s) { 1 }

int P(Person p) { 2 }

Student s = new Student();
Person p1 = new Person();
Person p2 = s;

P(s); 1

P(p1); 2

P(p2); 2

روز مقاومت اسلامی

Map (نقشه)
value و key
map < String, Integer > m = new HashMap<String, Integer>();
m.put("Ali", 12);
m.put("Ahmad", 24);
m.put("Saeed", 24);
m.put("Ali", 13); // replace previous
System.out.println("m.get('Ali')");

SortedMap (I) ← SortedMap (I)
HashMap
m.keySet() → for (String k: m.keySet())
System.out.println(m.get(k));
values will be written.
keys ← System.out.println(m.keySet());
values ← System.out.println(m.values());

```
public class main {
    public static void main(String[] args) {
        Collection<String> c = new ArrayList<String>();
        c.add("iman");
        c.add(new String("Ali"));
        true ← System.out.println(c.contains("iman"));
        c.remove(new String("iman"));
        false ← System.out.println(c.contains("iman"));
    }
}
```

۱۳۹۶ مرداد

۲۷ / دی القعدة / ۱۴۳۸

20 / August / 2017



۲۹
یکشنبه



collections

Inter Faces

Iterable

Collection

List

Queue

Set

Deque

SortedSet

classes

ArrayList

LinkedList

Vector

Stack

Priority Queue

ArrayDeque

HashSet

LinkedHashSet

TreeSet



بانک صادرات ایران

سه شنبه

nested

Inner Class

```
public class Outer {  
    private int a = 1;  
}
```

```
public class InnerS
{
    public void f()
    {
        a = 2;
        System.out.println("Fu in inner" + a);
    }
}

public void g()
{
    // ...
}
```

```
System.out.println("gc = " + a);
```

```
Inner In = new Inner();
```

In. $f()$;

```
System.out.println(gc after In.f0  
+ a);
```

public static class staticInner {
 }
 public void K() {
 System.out.println("K(): static Inner");
 }
 }

بهرام صیبری کی بجای دستنویس
 وزارت
 برای اعلام دستنویس از این عباس
 شهادت حضرت امام محمد باقر علیه السلام (۱۱۴۰ هـ)
 بهرام صیبری کی بجای دستنویس
 عباس وزارت

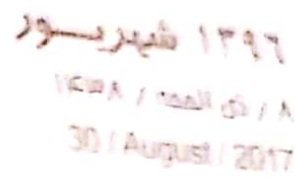
دوشنبه ۶

23 / August / 20

شهادت مطلوبہ الزمان حوالہ دست مأموران ال سعود (۱۳۶۶ھ ش - برابر با ۶ ذی الحجہ ۱۴۰۷ھ ق)



کتابخانه ملی و اسنادخانه ایران



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

 $0.9()$
$$\text{Inobj. f. c)} ;$$
 $\sin k(x)$

80: 1

Inner fl: 2



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

۱۳۹۶ شهریور

۹ / ذی الحجه / ۱۴۳۸

31 / August / 2017



۹
پنجشنبه



Inner class
implement, extend و ...
دا

outer
private string msg;
public string getMessage2() {
 msg = new Message();
 @Override
 public string getMessage() {
 return "This is an anonymous implementation";
 }
}

روز عرفه (روز نیايش)

۱۳۹۶ شهریور

۱۰ / ذی الحجه / ۱۴۳۸

1 / September / 2017



۱۰
جمعه



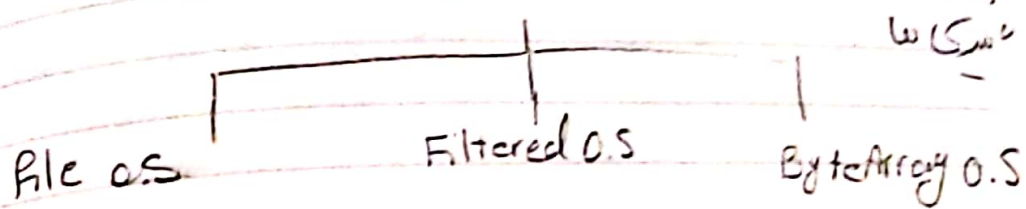
return msg.getMessage();

عبد سعید قربان (تعطیل) - روز بانکداری اسلامی



بانک صادرات ایران

out put Stream



نویسش
Writer, Reader
FileWriter FileReader

```

public class main {
    try {
        FileWriter fw = new FileWriter("a.txt");
        fw.write("salam\n");
        fw.write(02);
        fw.write("End");
    }
}

```

String
char array

→ eclips → \n
برای خط جدید
در اسلحه

```

        fw.close();
        FileReader fr = new FileReader("a.txt");
        int r = fr.read();
        System.out.println((char) r);
        fr.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

(*)



(*) همیشه بستن فایل در `finally` تکرار می‌شود تا در صورت داشتن
 اور فایل بسته شود.
`finally` }

`if (fw != null)`

البته باید `try` نوشته شود → `fw.close();` }

`fw` را در فایل از بدنه `try` تعریف می‌کنیم نه در `finally` تا فایل استفاده باشد.
 اگر بستن فایل به مشکل برخورد، باید `exception` بررسی نوشته شود.

فیلد (I) `Auto closable` → (I) `Closable` ← interface ←

```
try (FileWriter fw = new FileWriter("a.txt")) {
    ...
}
```

در نیازی به نوشتن `finally` است



۱۳
دوشنبه



۱۳۹۶ شهریور
۱۳ / ذی القعدة / ۱۴۳۸
4 / September / 2017

int a = 4;

binary file

```
try (FileOutputStream fos = new FileOutputStream  
("binaryFile"); ) {
```

```
    fos.write(50);
```

```
    fos.write(b); }
```

```
catch (FileNotFoundException e) {
```

```
    e.printStackTrace(); }
```

```
catch (IOException e) {
```

```
    e.printStackTrace(); }
```

```
try (FileInputStream fis = new FileInputStream  
("binaryFile"); ) {
```

```
    int readByte;
```

```
    while ( (readByte = fis.read()) != -1 ) {
```

```
        bytes.add(readByte);
```

```
        System.out.println(readByte);
```

←

روز تعاون - روز بزرگداشت ابو یحییٰ عروسی - روز مردم شناسی

بانک صادرات ایران



```
catch (FileNotFoundException e) {
```

```
e.printStackTrace();}
```

```
catch (IOException e) {
```

```
e.printStackTrace();}
```

```
System.out.println(bytes.size());
```

← برای استفاده از اعداد قبل از try

```
List<Byte> bytes = new ArrayList<>();
```

کلاس File

اطلاعات فایل را برمیگرداند و File f = new File("a.txt");

```
System.out.println(f.canRead());
```

```
System.out.println(f.exists());
```

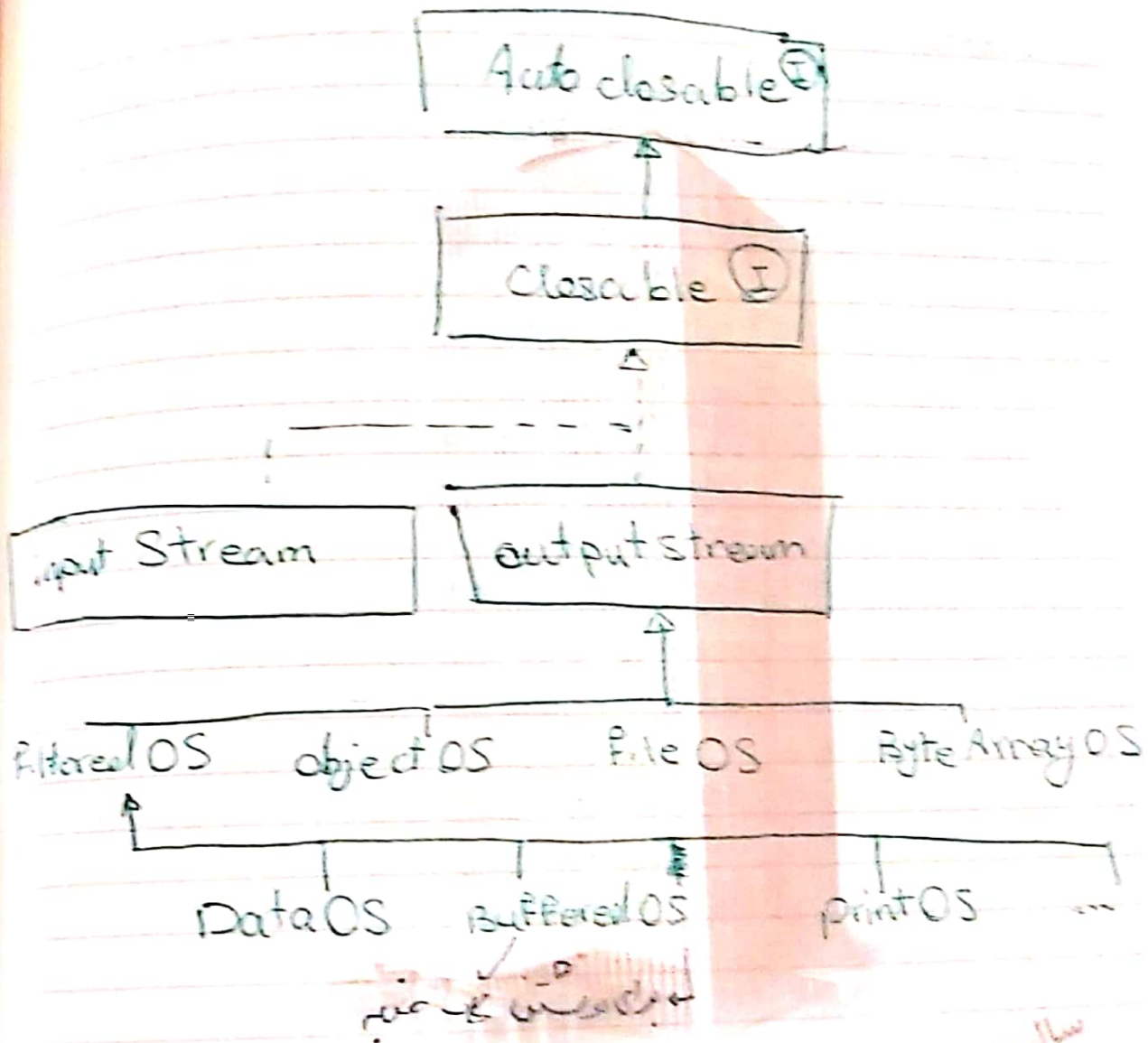
```
f.length()
```

```
f.getName()
```

```
f.getAbsolutePath()
```

```
f.isDirectory()
```





سالمروز قانسيس بانک صادرات ايران 15 شهر يور 1331 گرامی باد
65 سال قدمت 65 سال خدمت

۱۳۹۶ شهریور

۱۴ / ذی الحجه / ۱۴۳۸

7 / September / 2017



```
public class person implements Serializable {
```

```
    public String name;
```

```
    public int Age; دانشگاه ← public transient int age;
```

```
    public int[] grade; + getter + setter
```

main → →

```
FileOutputStream FFOS = new FileOutputStream  
("object File");
```

```
ObjectOutputStream objos = new ObjectOutputStream  
(FFOS);
```

۱۳۹۶ شهریور

۱۷ / ذی الحجه / ۱۴۳۸

8 / September / 2017



قیام ۱۷ شهریور و کشتار جمعی از مردم به دست مأموران ستم شاهی پهلوی (۱۳۵۷ هـ.ش)



سازمان اسناد و کتابخانه ملی جمهوری اسلامی ایران



۱۳۹۶ شهریور

۱۸ / ۱۴۳۸ / ۱۸

9 / September / 2017

Thread

Content switch ← سیستم عامل انجام می‌دهد. (اصلاً بین لسترل نیست)
پردازنده (Core) به اندازه دو پردازنده multi tasking

```
public class main {
```

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

```
        FirstThread ft = new FirstThread();
```

```
        ft.start();
```

Thread استیو از خط

```
        for (int i = 1000; i < 1100; i++)  
            System.out.println(i);
```

1100 ✗

```
        Thread t2 = new Thread(new Runnable  
            class() {
```

```
            public void run() {
```

```
                try {
```

```
                    t1.join();
```

```
                    t2.join();
```

```
                } catch (InterruptedException e) {
```

```
                    e.printStackTrace();
```

ت۱ و ت۲
Thread
ت۱ و ت۲

متوالیاتی که باید انجام شود



بسم الله الرحمن الرحیم



```
public class firstThread extends Thread {
```

```
    @Override
```

```
    public void run() {
```

```
        for (int i = 0; i < 100; i++)
```

```
            System.out.println(i);
```

```
public class RunnableClass implements Runnable
```

```
{
```

```
    @Override
```

```
    public void run() {
```

```
    }
```

وفات آیت الله سید محمود طالقانی اولین امام جمعه تهران (۱۳۵۸ هـ ش)



بانک صادرات ایران

معن = متوقف کردن Thread
Sleep (x) = توقف Thread
t.sleep(x)

② در Sleep (x) فقط x ثانیه متوقف می شود Thread به معنای متوقف می شود.

③ current Thread و متوقف است به Thread جاری که در دسترس است.
Thread.sleep(1000);

④ هر Thread می تواند متوقف شود.

System.out.println (t1.getId());

* برای اولویت بندی: `t1.set priority (Thread. max-priority);`

↓
10 1
max - min

* `t1.setae Demon (true);`
 متد `Set Demon` /
 کنترل و محدود کردن عمر Thread ها

* استفاده از Anonymous کلاس برای Thread
`Thread t2 = new Thread (new Runnable() {`

@override

`public void run() {`

`for (int i = 20000; i < 20100; i++)`

`system.out.println(i); }`

`t2.start();`

↑
main

* Synchronized block
public void increase() {
synchronized (this) {
gold ++;
}}

```
public class main {
    private int gold;
    public void increase() {
        gold ++;
    }
    public void decrease() {
        gold --;
    }
}
```

lock
۱۰,۰۰۰
قفل
ری استادی

@Override → toString

Thread ها به صورت موازی اجرا می شوند و به همین دلیل نیاز به کنترل داریم (wait) می ماند
در غیر این صورت به صورت متوالی اجرا می شوند و اولی می رفته

۱۳۹۶ شهریور
۱۳۳۸ / ذی الحجه / ۲۴
15 / September / 2017

```
public class worker implements Runnable {
    Coin coin;
    @Override
    public void run() {
        coin.increase();
        coin.increase();
        Thread.currentThread().sleep(50);
        coin.decrease();
    }
}
```

try catch + Thread.currentThread().sleep(50);

```
public class main {
    public static void main(String[] args) {
        Coin c = new Coin();
        List<Thread> threads = new
        ArrayList<>();
        for (int i = 0; i < 100; i++) {
            Thread t = new Thread(new worker(c));
            threads.add(t);
            t.start();
        }
        for (int i = 0; i < 100; i++) {
            try catch + threads.get(i).join();
            System.out.println(c);
        }
    }
}
```

Race condition
gold +1 +2
critical section
lock

Thread ← interrupted exception
باید روی حواشی
interrupt ← exception
باید حواشی را برد - باید exception

شهریور ۱۳۹۶
۲۸ / ذی الحجه / ۱۴۳۸
19 / September / 2017



Reflection (بازتاب) Runtime

class object

field
method
Constructor
:

Object در این مرحله به صورت واقعی می آید

class → obj نور مرئوس

Class c = s.getClass

getClass در این مرحله به صورت واقعی می آید

class c = String.class

className

try catch class c = Class.forName("java.lang.String");

* object o = new String();

if (o instanceof String) {

}

if (o.getClass().getName().equals("java.lang.String"))

{

}



if (String.class.isInstance(o))

{
}

Field

اطلاعات متعلق به کلاس را می توان به صورت

```
public class person {
```

```
    private int age;
```

```
    private String name;
```

```
    public person(int age, String name) {
```

```
        this.age = age;
```

```
        this.name = name;
    }
```

```
    public person ()
```

```
    {
```

```
    }
```

```
person p = new person(20, "ali");
```

```
p = p.getClass().getFields();
```

Field[] fields

۱۳۹۶ شهریور

۱۳۹۶ / مهر / ۲۰

21 / September / 2017



Student, Class, getFields

```
for (Field f : fields) {
```

```
    System.out.println(f.getName());
```

```
}
```

public / private

Field[] decFields = ^{person} Student.class.getDeclaredFields

```
for (Field f : decFields) {
```

```
    System.out.println(f.getName());
```

```
}
```

۱۳۹۶ شهریور

۱۳۹۶ / مهر / ۲۱

22 / September / 2017



ایمیل: ۱۳۹۶ شهریور / ایمیل: ۱۳۹۶ شهریور / ایمیل: ۱۳۹۶ شهریور





23 / September / 2017

}

Q.10. Class . getMethod() - getDeclaredMethod()

```
meth = Person.class.getMethod("setName", String.class);
```

method $\rightarrow (p, \text{"hamid"})$

meth. invoke (obj, Arg)

Static بودن null میسر.



بانک صادرات ایران

try {



یکشنبه



or (int.class, String.class)

Constructo pc = person.class.getConstructor();

Object o = pc.newInstance(); or (23, "Taghi")

System.out.println(o.toString());

Remote procedure call

← Rpe

لے استعاره



بانک صادرات ایران



۱۳۹۶ شهریور
۱۸ / دی ماه / ۱۴۳۸
9 / September / 2017

Thread

Content switch ← سیستم عامل انجام می دهد. (اصلاً قابل کنترل نیست)

پردازش همزمان (Core) به اندازه ۲ یا ۴ هسته Multi tasking

```
public class main {
```

```
    public static void main ( ) {
```

```
        FirstThread ft = new FirstThread();
```

```
        ft.start();
```

Thread

ارتباطی با

۱۱۰۰

```
        for (int i = 1000; i < 1010; i++)  
            System.out.println(i);
```

```
        Thread t2 = new Thread (new Runnable  
                                class () {
```

```
            public void run () {
```

```
                try {
```

```
                    t1.join();
```

```
                    t2.join();
```

```
                } catch (InterruptedException e) {
```

```
                    e.printStackTrace();
```

```
                }
```

* متغیر e از نوع Exception است



بسم الله الرحمن الرحيم