Java OOP
10128

Inheritance

Pini shlomi

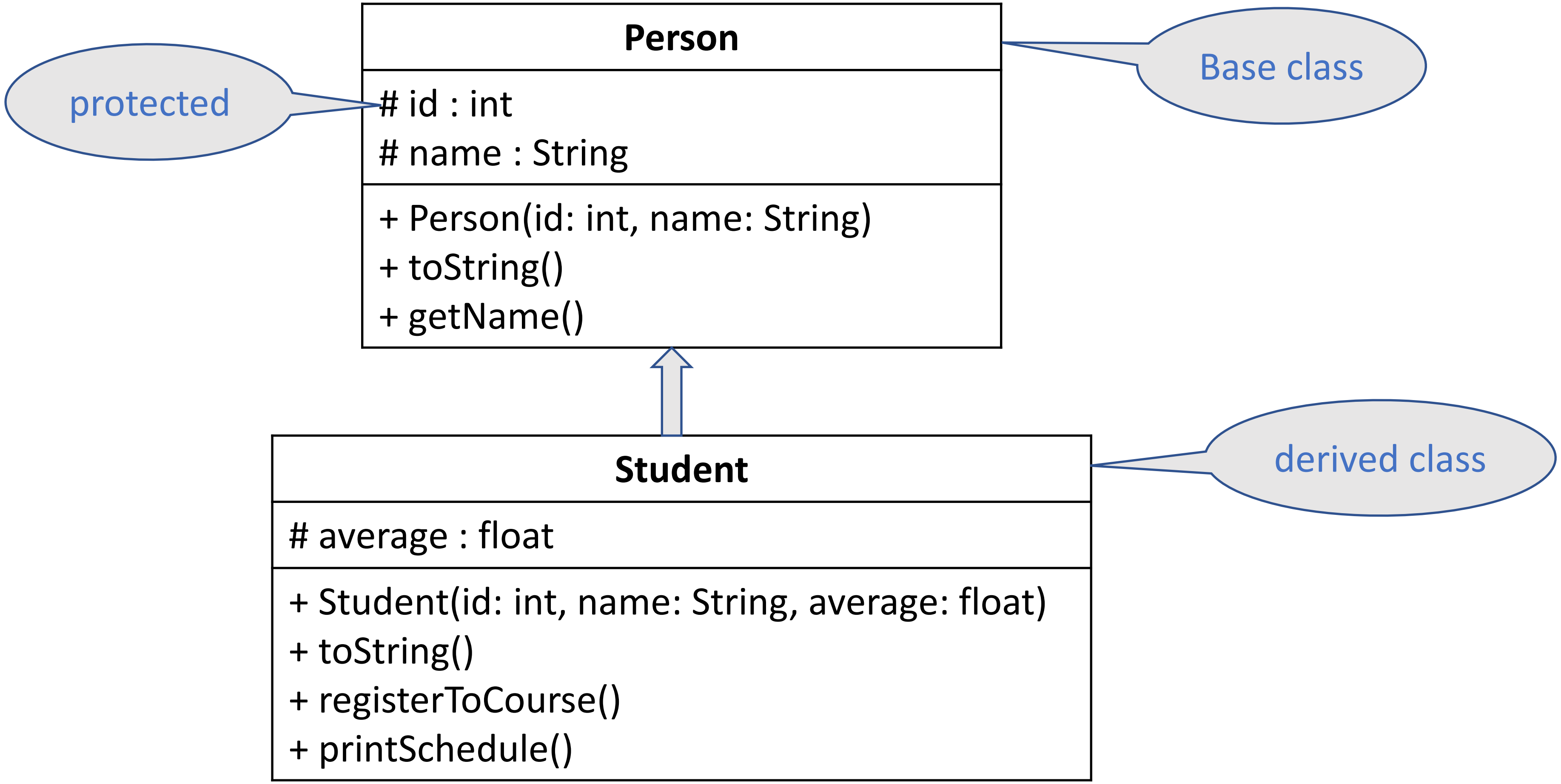
Object class – base methods

- tostring()
- hashCode()
- equals(Object obj)
- finalize()
- getClass()
- clone()
- wait()
- notify()
- notifyAll()

more info

Inheritance

In Java we can inherited from only one class



protected

Person

# id : int
# name : String

+ Person(id: int, name: String)
+ toString()
+ getName()

Base class

Student

# average : float

+ Student(id: int, name: String, average: float)
+ toString()
+ registerToCourse()
+ printSchedule()

derived class

Inheriting class access permission

- Inheriting class containing all parent attributes / methods
- But can't access to private attributes / methods
- Using protected permission solve that problem

Protected permission:

- In class
- In inheriting classes
- In classes that in same package

Base class

protected attributes

```
public class Person {

  protected int id;
  protected String name;

  public Person(int id, String name) {
    this.id = id;
    this.name = name;
  }

  public String getName() {
    return name;
  }

  @Override
  public String toString() {
    return id + ", " + name ;
  }

  public void haveFun() {
    System.out.println("wondering what to do!");
  }
}
```

Inheritance class

```
public class Student extends Person {

  protected float average;

  public Student(int id, String name, float average) {
    super(id, name);
    this.average = average;
  }

  @Override
  public String toString() {
    return super.toString() + ", " + average;
  }

  @Override
  public void haveFun() {
    System.out.println("Doing homework");
  }
}
```

Inheritance

Call parent constructor

Override parent method

Override parent methods

```
public class NormalStudent extends Student {

  public NormalStudent(int id, String name, float average) {
    super(id, name, average);
  }

  @Override
  public void haveFun() {
    System.out.println("Playing...");
  }
}
```

Call parent method

Main

```
public static void main(String[] args) {
  Student s1 = new Student(25, "Ram", 97.5f);
  System.out.println(s1);
  s1.haveFun();
  NormalStudent ns1 = new NormalStudent(19, "Yosi", 90f);
  System.out.println(ns1);
  ns1.haveFun();
}
```

Console

```
25, Ram, 97.5
Doing homework
19, Yosi, 90.0
Playing...
```

Final class and method

Can’t inheriting final class

```
final class A {
}

class B extends A {
}
```

The type B cannot subclass the final class A

Can’t override final method

```
class C{
  public final void f() {
  }
}

class D extends C {
  @Override
  public void f() {
  }
}
```

Cannot override the final method from C

תרגיל 1 : מה יקרה בהרצת הקוד הבא?

```java
class Person {
    protected int id;
    protected String name;

    public Person(int id, String name) {
        this.id = id;
        this.name = name;
    }

    public void haveFun() {
        System.out.println("wondering what to do!");
    }
}

class Student extends Person {
    protected float average;

    public Student(int id, String name, float average) {
        super(id, name);
        this.average = average;
    }

    @Override
    public void haveFun() {
        System.out.println("Doing homework");
    }
}

public class Main {
    public static void main(String[] args) {
        Student s = new Student(1, "Dana", 95.5f);
        s.haveFun();
    }
}
```

תרגיל 2 : מה יקרה בהרצת הקוד הבא?

```
class Person {
   protected int id;
   protected String name;

   public Person(int id, String name) {
      this.id = id;
      this.name = name;
   }

   @Override
   public String toString() {
      return id + ", " + name;
   }
}

class Student extends Person {
   protected float average;

   public Student(int id, String name, float average) {
      super(id, name);
      this.average = average;
   }
   @Override
   public String toString() {
      return super.toString() + ", " + average;
   }
}

public class Main {
   public static void main(String[] args) {
      Student s = new Student(2, "Noa", 88.8f);
      System.out.println(s);
   }
}
```

תרגיל 3 : מה יקרה בהרצת הקוד הבא?

```
class A {
   protected int id;

   public A(int id) {
      this.id = id;
   }

   public final void f() {
      System.out.println("Final method");
   }
}

class B extends A {
   private String value;
   public B(int id, String value) {
      super(id);
      this.value = value;
   }

@Override
   public void f() {
      System.out.println("Trying to override");
   }
@Override
   public String toString() {
      return super.toString() + ", value: " + value;
   }
}

public class Main {
   public static void main(String[] args) {
      A a = new A(5);
      System.out.println(a);
      A b = new B(5, "First Value");
      System.out.println(a);
   }
}
```

תרגיל 4 : מה יקרה בהרצת הקוד הבא?

```
class Person {
   protected String name;

   public Person(String name) {
      this.name = name;
   }
}

class Student extends Person {
   public Student(String name) {
      super(name);
   }

   public void printName() {
      System.out.println(name);
   }
}
```

```
public class Main {
   public static void main(String[] args) {
      Student s = new Student("Gil");
      s.printName();
   }
}
```

תרגיל 5 : מה יקרה בהרצת הקוד הבא?

```
class A {
   private int x = 5;

   public A(int x) {
      this.x = x;
   }
}

class B extends A {
   private final int y;
   public B(int x, int y) {
      this.y = y;
   }

   public void printX() {
      System.out.println(y);
   }
}
```

```
public class Main {
   public static void main(String[] args) {
      B b = new B(25, 10);
      b.printX();
   }
}
```

תרגיל 6 : מה יקרה בהרצת הקוד הבא?

```java
class Person {
    public void haveFun() {
        System.out.println("Generic fun");
    }
}

class Student extends Person {
    @Override
    public void haveFun() {
        System.out.println("Student fun");
    }
}
```

```java
public class Main {
    public static void main(String[] args) {
        Person p = new Student();
        p.haveFun();
    }
}
```

תרגיל 7 : מה יקרה בהרצת הקוד הבא?

```
class Person {
   public void haveFun() {
      System.out.println("Parent fun");
   }
}

class Student extends Person {
   @Override
   public void haveFun() {
      super.haveFun();
      System.out.println("Student fun");
   }
}
```

```
public class Main {
   public static void main(String[] args) {
      Student s = new Student();
      s.haveFun();
   }
}
```

תרגיל 8 : מה יקרה בהרצת הקוד הבא?

```
class Person {
   protected String name;

   public Person(String name) {
      this.name = name;
   }

   public void printName() {

System.out.println(name.toUpperCase());
   }
}

class Student extends Person {
   public Student(String name) {
      super(name);
   }
}
```

```
public class Main {
   public static void main(String[] args) {
      Student s = new Student(null);
      System.out.println("Before print name:");
      s.printName();
      System.out.println("After print name");
   }
}
```

תרגיל 9 : מה יקרה בהרצת הקוד הבא?

```
class Person {
   protected String name;

   public Person(String name) {
      this.name = name;
   }

   public void sayHello() {
      System.out.println("Hello, " + name);
   }
}
```

```
public class Main {
   public static void main(String[] args) {
      Person[] people = new Person[2];
      people[0] = new Person("Alice");
      people[1] = new Person("Bob");

      for (int i = 0; i <= 2; i++) {
         System.out.println("Index: " + i);
         people[i].sayHello();
      }
   }
}
```

תרגיל חברת תוכנה:

קישור לתרגיל ניהול חברת תוכנה

הורדת הקוד ההתחלתי:

כנסו לאתר בקישור הבא: אתר להורדת ספריות מ-github.

באתר שימו את ה-url הבא:

https://github.com/pinishlomi/java_oop_exercises/tree/master/src/starters/programing_company