BJP4 Exercise 9.4: MonsterTruck

Language/Type: Java classes implementing inheritance instance methods
Related Links: Car.java
Author: Marty Stepp (on 2016/09/08)

Suppose that the following two classes have been declared:

```
public class Car {
    public void m1() {
        System.out.println("car 1");
    }

    public void m2() {
        System.out.println("car 2");
    }

    public String toString() {
        return "vroom";
    }
}

public class Truck extends Car {
    public void m1() {
        System.out.println("truck 1");
    }

    public void m2() {
        super.m1();
    }

    public String toString() {
        return super.toString() + super.toString();
    }
}
```

Write a class `MonsterTruck` whose methods have the behavior below. Don't just print/return the output; whenever possible, use inheritance to reuse behavior from the superclass.

Method	Output/Return
m1	monster 1
m2	truck 1 car 1
toString	"monster vroomvroom"

Type your solution here:

```
1  public class MonsterTruck extends Truck {
2      public void m1() {
3          System.out.println("monster 1");
4      }
5      public void m2() {
6          super.m1();
7          super.m2();
8
9      }
10     public String toString() {
11         return "monster " + super.toString();
12     }
13
14
15 }
```

This is an **inheritance problem.** Write a Java class using inheritance. (You do not need to write any `import` statements.)

4 Indent

Sound F/X

Highlighting

Submit

You passed 3 of 3 tests.

Go to the next problem: MinMaxAccount

test #1:	m1
console output:	monster 1
result:	pass
test #2:	m2
console output:	truck 1 car 1
result:	pass
test #3:	toString

console output:
```
"monster vroomvroom"
"monster vroomvroom"
```

result: pass

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.
If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

thle01

< Janitor | Main Page → Problems → Solve a Problem | MonsterTruck >

Show Header

BJP4 Exercise 9.3: HarvardLawyer

Language/Type: Java classes implementing inheritance instance methods
Related Links: Lawyer.java
Author: Marty Stepp (on 2016/09/08)

Write a class `HarvardLawyer` to accompany the other law firm classes described in this chapter. Harvard lawyers are like normal lawyers, but they make 20% more money than a normal lawyer, they get 3 days more vacation, and they have to fill out four of the lawyer's forms to go on vacation. That is, the `getVacationForm` method should return `"pinkpinkpinkpink"`. Make sure to interact with the superclass as appropriate.

Type your solution here:

```
1  public class HarvardLawyer extends Lawyer {
2      public double getSalary() {
3          return super.getSalary() * 1.2;
4      }
5      public int getVacationDays() {
6          return super.getVacationDays() + 3;
7      }
8      public String getVacationForm() {
9          return super.getVacationForm()+super.getVacationForm()
10             +super.getVacationForm()+super.getVacationForm();
11     }
12
13 }
```

This is an **inheritance problem.** Write a Java class using inheritance. (You do not need to write any `import` statements.)

4 Indent

Sound F/X

Highlighting

Submit

You passed 7 of 7 tests.

Go to the next problem: MonsterTruck

test #1: `getSalary 1`

console output:	`48000.0`
result:	pass

test #2:	`getSalary 2 (change base salary to $56000.00)`
console output:	`67200.0`
result:	pass

test #3:	`getVacationDays 1`
console output:	`18`
result:	pass

test #4:	`getVacationDays 2 (change base days to 18)`
console output:	`26`
result:	pass

test #5:	`getVacationForm 1`
console output:	`"pinkpinkpinkpink"`
result:	pass

test #6:	`getVacationForm 2 (change Lawyer form to green)`
console output:	`"greengreengreengreen"`
result:	pass

test #7:	`sue`
console output:	`I'll see you in court!`
result:	pass

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.
If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

thle01

< Marketer Main Page → Problems → Solve a Problem HarvardLawyer >

Show Header

BJP4 Exercise 9.2: Janitor

Language/Type: Java classes implementing inheritance instance methods
Related Links: Employee.java
Author: Marty Stepp (on 2016/09/08)

Write a class `Janitor` to accompany the other law firm classes described in this chapter. Janitors work twice as many hours per week as other employees (80 hours/week), they make $30,000 ($10,000 less than general employees), they get half as much vacation as other employees (only 5 days), and they have an additional method clean that prints `"Workin' for the man."` Make sure to interact with the superclass as appropriate.

Type your solution here:

```
1  public class Janitor extends Employee {
2      public double getSalary() {
3          return super.getSalary() - 10000.0;
4      }
5      public int getHours() {
6          return super.getHours() * 2;
7      }
8      public int getVacationDays() {
9          return super.getVacationDays() / 2;
10     }
11     public void clean() {
12         System.out.println("Workin' for the man.");
13     }
14
15 }
```

This is an **inheritance problem.** Write a Java class using inheritance. (You do not need to write any `import` statements.)

4 Indent

Sound F/X
Highlighting

Submit

You passed 7 of 7 tests.

Go to the next problem: HarvardLawyer

test #1:	`getSalary 1`
console output:	`30000.0`
result:	pass
test #2:	`getSalary 2 (change base salary to $65535.00)`
console output:	`55535.0`
result:	pass
test #3:	`getHours 1`
console output:	`80`
result:	pass
test #4:	`getHours 2 (change base hours to 36)`
console output:	`72`
result:	pass
test #5:	`getVacationDays 1`
console output:	`5`
result:	pass
test #6:	`getVacationDays 2 (change base days to 18)`
console output:	`9`
result:	pass
test #7:	`clean`
console output:	`Workin' for the man.`
result:	pass

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.
If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

thle01

< CarTruck2 Main Page → Problems → Solve a Problem Janitor >

Show Header

BJP4 Exercise 9.1: Marketer

Language/Type: Java classes implementing inheritance instance methods
Related Links: Employee.java
Author: Marty Stepp (on 2016/09/08)

Write the class `Marketer` to accompany the other law firm classes described in this chapter. Marketers make $50,000 ($10,000 more than general employees) and have an additional method called `advertise` that prints `"Act now, while supplies last!"` Make sure to interact with the `Employee` superclass as appropriate.

Type your solution here:

```
1 public class Marketer extends Employee {
2     public double getSalary() {
3         return super.getSalary() + 10000.0;
4     }
5     public void advertise() {
6         System.out.println("Act now, while supplies last!");
7     }
8 }
```

This is an **inheritance problem.** Write a Java class using inheritance. (You do not need to write any `import` statements.)

4 Indent

Sound F/X
Highlighting

Submit

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.
If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

< CarTruck Main Page → Problems → Solve a Problem Marketer >

BJP4 Self-Check 9.9: CarTruck2

Language/Type: Java classes inheritance mystery
Author: Marty Stepp (on 2016/09/08)

Consider the following two classes:

```
public class Car {
    public void m1() {
        System.out.println("car 1");
    }

    public void m2() {
        System.out.println("car 2");
    }

    public String toString() {
        return "vroom";
    }
}

public class Truck extends Car {
    public void m1() {
        System.out.println("truck 1");
    }

    public void m2() {
        super.m1();
    }

    public String toString() {
        return super.toString() + super.toString();
    }
}
```

And assuming that the following variable has been declared:

```
Truck mytruck = new Truck();
```

What is the output from the following statements?

```
System.out.println(mytruck);
mytruck.m1();
mytruck.m2();
```

```
vroomvroom
truck 1
car 1
```

Sound F/X

Submit

You passed 3 of 3 tests.

Go to the next problem: Marketer

#	question	your answer	result
1	`System.out.println(mytruck);`	`vroomvroom`	pass
2	`mytruck.m1();`	`truck 1`	pass
3	`mytruck.m2();`	`car 1`	pass

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.
If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

thle01

< inheritanceVariableSyntax Main Page → Problems → Solve a Problem CarTruck2 >

BJP4 Self-Check 9.8: CarTruck

Language/Type: Java classes inheritance mystery
Author: Marty Stepp (on 2016/09/08)

Consider the following two classes:

```
public class Car {
    public void m1() {
        System.out.println("car 1");
    }

    public void m2() {
        System.out.println("car 2");
    }

    public String toString() {
        return "vroom";
    }
}

public class Truck extends Car {
    public void m1() {
        System.out.println("truck 1");
    }
}
```

And assuming that the following variables have been declared:

```
Car mycar = new Car();
Truck mytruck = new Truck();
```

What is the output from the following statements?

`System.out.println(mycar);`	`vroom`
`mycar.m1();`	`car 1`
`mycar.m2();`	`car 2`
`System.out.println(mytruck);`	

Sound F/X

```
mytruck.m1();
mytruck.m2();
```

```
vroom
truck 1
car 2
```

Submit

You passed 6 of 6 tests.

Go to the next problem: CarTruck2

#	question	your answer	result
1	`System.out.println(mycar);`	`vroom`	pass
2	`mycar.m1();`	`car 1`	pass
3	`mycar.m2();`	`car 2`	pass
4	`System.out.println(mytruck);`	`vroom`	pass
5	`mytruck.m1();`	`truck 1`	pass
6	`mytruck.m2();`	`car 2`	pass

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.
If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

thle01

< subclassSyntax Main Page → Problems → Solve a Problem CarTruck >

BJP4 Self-Check 9.10: inheritanceVariableSyntax

Language/Type: Java classes inheritance syntax variables
Author: Marty Stepp (on 2016/09/08)

Consider the following classes:

```
public class Vehicle {...}
public class Car extends Vehicle {...}
public class SUV extends Car {...}
```

Which of the following are legal statements?

Sound F/X

a. ☐ Car c = new Vehicle();
b. ☑ Car c = new SUV();
c. ☐ SUV s = new Car();
d. ☑ Vehicle v = new Car();
e. ☑ SUV s = new SUV();
f. ☑ Vehicle v = new SUV();
(order shuffled)

Submit

You passed 1 of 1 tests.

Go to the next problem: CarTruck

question #1: Which of the following are legal statements?
your answer: Car c = new SUV();
Vehicle v = new Car();
SUV s = new SUV();
Vehicle v = new SUV();
result: pass

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.
If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

thle01

Main Page → Problems → Solve a Problem inheritanceVariableSyntax >

BJP4 Self-Check 9.3: subclassSyntax

Language/Type: Java classes inheritance syntax
Author: Marty Stepp (on 2016/09/08)

Which of the following is the correct syntax to indicate that class `A` is a subclass of `B`?

Sound F/X

a. `public class B extends A {`
b. `public class A : super B {`
c. `public A implements B {`
d. `public A(super B) {`
e. `public class A extends B {`

(order shuffled)

Submit

You passed 1 of 1 tests.

Go to the next problem: inheritanceVariableSyntax

question #1:	Which of the following is the correct syntax to indicate that class `A` is a subclass of `B`?
your answer:	`public class A extends B {`
result:	pass

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.

If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.