

Computer Science AP

Polymorphism Worksheet

These are questions associated with each video in the Polymorphism Section.

Questions for Video: Introduction to Polymorphism 01

Use the project InheritanceExamples for these questions.

Consider the StudentTeacher01 package.

State *True* or *False* as to whether or not the following lines would cause a compile time error.

- ___ Student S=new Student("Adam", 15, 123)
- ___ Student S=new Person("Adam", 15)
- ___ Person P=new ExchangeStudent("Adam", 15, 123)
- ___ ExchangeStudent S=new Student("Adam", 15, 123)
- ___ Person P=new ExchangeStudent("Adam", 15, 123)
- ___ ExchangeStudent S=new Person("Adam", 15)
- ___ Person P=new Teacher("John", 45)
- ___ Teacher T=new Person("John", 45)

Consider the LCDScreen package.

State *True* or *False* as to whether or not the following lines would cause a compile time error.

- ___ Screen S=new ColorScreen("a83jf")
- ___ ColorScreen S=new Screen("fji8d")
- ___ ColorScreen S=new ColorScreenPlus("sif8d", 123)
- ___ ColorScreenPlus S=new Screen("ajf8d")
- ___ Screen S=new ColorScreenPlus("ji8de", 123)

A student creates a project that consists of exactly three classes. The three classes make use of inheritance through the use of 'extends'. The following lines compile without error. Can you determine which classes extend which classes? Explain.

```
Goobly G = new FaFoo()  
Babber B = new FaFoo()  
Goobly G = new Babber()
```

Questions for Video: Introduction to Polymorphism 01

There were several rules about what the compiler will allow when it comes to reference fields calling methods on instances that may or may not be subclasses of the reference field type (example: `Student S = new ExchangeStudent("Adam", 15, 123)` The instance in memory is not of the same type as the field type `Student`). Use the `InheritanceExamples01` project to answer the following questions to check whether or not you have a good understanding about these rules! Remember there is a logic as to WHY the rule is in place. Understand the logic and the rule should be easier to learn.

Use the `StudentTeacher01` package for these questions.

`Person P=new Student("Adam", 15,123)`
Which lines will cause a compile time error?

- `P.talk()`
- `P.setAge(15)`
- `P.study()`
- `double g=P.getAverageMark()`
- `P.stuff()`

`Student S=new ExchangeStudent("Adam", 15, 123)`
Which lines will cause a compile time error?

- `S.talk()`
- `S.stuff()`
- `double d=S.getAverageMark()`
- `double d=S.getUnadjustedAverageMark()`
- `String s=S.getReturnDate()`

Consider the concept in polymorphism that you can cast instances in memory into other types if required and when acceptable. Which of the following are acceptable casts?

<code>Pesron P=new ExchangeStudent("Adam", 15, 123)</code> <code>((ExchangeStudent)P).getReturnDate()</code>	
<code>Person P=new ExchangeStudent("Adam", 15, 123)</code> <code>((Student)P).study()</code>	
<code>Person P=new Student("Adam", 15, 123)</code> <code>((ExchangeStudent)P).getReturnDate()</code>	
<code>Student S=new Student("Adam", 15, 123)</code> <code>((Person)S).stuff()</code>	
<code>Student S=new Student("Adam", 15, 123)</code> <code>((ExchangeStudent)S).getReturnDate()</code>	
<code>Person P = new Teacher("John", 45)</code> <code>((Teacher)P).greet(Adam)</code> <code>//assume Adam is a Person...</code>	

Indicate whether the following code segments are good, have compile time errors (detected by the IDE), or run time errors (will cause error when running). For segments with errors you should clearly indicate the reason for the error. Use the LCDScreen package for code reference.

<pre>Screen S=new ColorScreen("DC5") S.display("Hello")</pre>	
<pre>Screen S=new ColorScreen("DC5") S.setColor(Color.BLACK) //Color.Black is an acceptable Color...</pre>	
<pre>Screen S=new ColorScreen("DC5") ((ColorScreenPlus)S).useSpecialEffect(3)</pre>	
<pre>Screen S=new ColorScreen("DC5") ((ColorScreen)S).useSpecialEffect(3)</pre>	
<pre>ColorScreen CS=new ColorScreen("DC5") CS.setVertical(150)</pre>	

Questions for Video: Polymorphism in Action 01 and 02

Since a reference type can possibly reference more than one type of object in memory, more possibilities open up when it comes to method arguments and method return types. Here are some questions to check if you picked up the basics on this topic.

Use the InheritanceExamples project, StudentTeacher01 package for these questions.

```
Teacher T=new Teacher("John", 45)
T.greet ( ??? )
T.makeStudy( ??? )
```

What type/s of instances from this project could you pass as an argument into the Teacher's greet method?

What type/s of instances from this project could you pass as an argument into the Teacher's makeStudy method?

```
Student Adam=new Student("Adam", 15, 123)
ExchangeStudent Bob=new ExchangeStudent("Bob", 15, 456)
Adam.partnerUp(Bob)
```

Will partnerUp method accept Bob as an argument? Explain.

A runner program uses the following lines that use the Teacher class' pickStudent method. Will the following lines compile and run? Explain. Assume that T is a teacher.

```
Student volunteer = T.pickStudent()
ExchangeStudent someone = T.pickStudent()
Person temp = T.pickStudent()
```

Assume that we add a method to the Student class has a method with the signature
public Person getFavoritePerson()
Could this method return an instance of type Student? ExchangeStudent? Teacher? Explain.

Use the InheritanceExamples project, LCDScreen package for these questions.

A runner program has an ArrayList of type Screen. The list contains several items.
`ArrayList<Screen> screens = new ArrayList<Screen>()`

What type/s of instances can be placed in this list?

For this question assume that all the screens in the list are either ColorScreens or ColorScreenPlus. Would the following line work in the program? Explain.
`ColorScreen temp = screens.get(0)`

What code would grab the very first item in the list and ask it to display "Hello!" ?

Assume that you don't know whether the first item in the list is a Screen, ColorScreen, or ColorScreenPlus. Will the line of code you wrote for the question above work regardless of the screen type? Explain why or why not.

What is the role of the operator 'instanceof'?

What code would grab the very first item in the list, check to see if it is a ColorScreenPlus, and if it is a ColorScreenPlus ask it to use special effect code 5.