|  |  |  |
| --- | --- | --- |
| **APCS Exposure Java** | **Exercises 14.01-10** | **Date:** |
| **Name:** | | **Period:** |

1. What is the opposite of *abstract*?

2. What is the primary goal of OOP?

3. How is reliability acquired through *class interaction*?

4. What kind of relationship does *inheritance* use?

5. What kind of relationship does *composition* use?

6. What does *polymorphism* mean?

7. Explain why the plus sign ( + ) is considered an *overloaded operator*?

8. Look at program ***Java1403.java***. This program has 4 different **drawSquare** methods.

How is this possible?

9. Look at ***Figure 14.7***. Which class is the *Umbrella Class* in this example?

10. The concept of the \_\_\_\_\_\_\_ \_\_\_\_\_\_\_is the foundation of polymorphism.

11. Compare programs ***Java1404.java*** and ***Java1405.java***.

What is different about the **English**, **German**, **Dutch**, and **French** classes in the second program?

12. What is the *Umbrella Class* in program ***Java1405.java***?

13. Explain how program ***Java1405.java*** is using *polymorphism*.

14. Explain how *polymorphism* improves the *Train Case Study*. (See ***Train.java*** in folder ***Java1406***.)

15. What does *API* stand for?

16. Explain how the **step** method of the **ActorWorld** class in GridWorld uses *polymorphism*.

17. What are the 4 **Collection** classes in Java?

18. What is a **Collection**?

19. What method is used to add data to a **Collection** object?

20. What is a **List**?

21. Can *lists* have duplicate elements?

22. What is a **Set**?

23. Compare programs ***Java1410.java*** & ***Java1411.java***. What change was made to the 2nd program?

24. Does program ***Java1411.java*** compile?

25. What is an **interface**?

26. What kind of methods are put into an **interface**?

27. Look at program ***Java1414.java***. What is wrong with this program?

28. If an interface has 5 **abstract** methods, how many of them must be "implemented" if a class implements the **interface**?

29. If a **class** implements an **interface**, can you define additional methods in the class that were not declared in the interface?

30. Can one **class** be implemented from multiple **interface**s?

31. Can data be stored in an **interface**? If "yes" what kind?

32. When declaring a *constant* in an **interface**, is the word **final** necessary?

33. Can an **interface** be used instead of an *umbrella class* for *polymorphism*?

34. Compare programs ***Java1417.java*** & ***Java1420.java***.

Is there any difference in the execution? If "yes" what?

35. Refer to the previous question.

Is there any difference in the code? If "yes" what?

36. Can data be stored in an **abstract class**? If "yes" what kind?

37. Is the keyword **abstract** required for an **abstract interface** and its **abstract** methods**?**

38. Is the keyword **abstract** required for an **abstract class** and its **abstract** methods**?**

39. Can an **interface** or an **abstract class** have a *constructor*?

40. When declaring a *constant* in an **abstract class**, is the word **final** necessary?

41. Can an **abstract class** be used to facilitate *polymorphism*?