1.Which of the following declarations does not compile?

A. double num1, int num2 = 0;

B. int num1, num2;

C. int num1, num2 = 0;

D. int num1 = 0, num2 = 0;

A.Option A does not compile because Java does not allow declaring different types as part of the same declaration. The other three options show various legal combinations of com- bining multiple variables in the same declarations with optional default values.

2.What is the output of the following?

```
public static void main(String... args)
{
String chair, table = "metal";
chair = chair + table;
System.out.println(chair);
}
```

A. metal

B. metalmetal

C. nullmetal

D. The code does not compile.

D. The table variable is initialized to "metal". However, chair is not initialized. In Java, initialization is per variable and not for all the variables in a single declaration. Therefore, the second line tries to reference an uninitialized local variable and does not compile, which makes Option D correct.

3.Which is correct about an instance variable of type String?

A. It defaults to an empty string.

B. It defaults to null.

C. It does not have a default value.

D. It will not compile without initializing on the declaration line.

B.Instance variables have a default value based on the type. For any non-primitive, includ- ing String, that type is a reference to null. Therefore Option B is correct. If the variable was a local variable, Option C would be correct.

4.Which of the following is not a valid variable name?

A. _blue

B. 2blue

C. blue$

D. Blue

C. An identifier name must begin with a letter, $, or _. Numbers are only permitted for subsequent characters. Therefore, Option B is not a valid variable name.

5.Which best describes what the new keyword does?

A. Creates a copy of an existing object and treats it as a new one

B. Creates a new primitive

C. Instantiates a new object

D. Switches an object reference to a new one

C. The new keyword is used to call the constructor for a class and instantiate an instance of the class. A primitive cannot be created using the new keyword. Dealing with references happens after the object created by new is returned.

6.Which is the first line to trigger a compiler error?

```
double d1 = 5f; // p1
double d2 = 5.0; // p2
float f1 = 5f; // p3
float f2 = 5.0; // p4
```

A. p1

B. p2

C. p3

D. p4

D. Java uses the suffix f to indicate a number is a float. Java automatically widens a type, allowing a float to be assigned to either a float or a double. This makes both lines p1 and p3 compile. Line p2 does compile without a suffix. Line p4 does not compile without a suffix and therefore is the answer.

7.Which of the following lists of primitive types are presented in order from smallest to largest data type?

A. byte, char, float, double
B. byte, char, double, float
C. char, byte, float, double
D. char, double, float, bigint

A.A byte is smaller than a char, making Option C incorrect. bigint is not a primitive, making Option D incorrect. A double uses twice as much memory as a float variable, therefore Option A is correct.

8.Which of the following is not a valid order for elements in a class?

A. Constructor, instance variables, method names
B. Instance variables, constructor, method names
C. Method names, instance variables, constructor
D. None of the above: all orders are valid.

D. The instance variables, constructor, and method names can appear in any order within a class declaration.

9.Which of the following lines contains a compiler error?

```
String title = "Weather";   // line x1
int hot, double cold; // line x2
System.out.println(hot + " " +title); // line x3
```

A. x1
B. x2
C. x3
D. None of the above

B.Java does not allow multiple Java data types to be declared in the same declaration, making Option B the correct answer. If double was removed, both hot and cold would be the same type. Then the compiler error would be on x3 because of a reference to an unini- tialized variable.

10.Of the types double, int, and short, how many could fill in the blank to have this code output 0?

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

```
defaultValue;
System.out.println(defaultValue);
}
```

A. None

B. One

C. Two

D. Three

A.Since defaultValue is a local variable, it is not automatically initialized. That means the code will not compile with any type. Therefore, Option A is correct. If this was an instance variable, Option C would be correct as int and short would be initialized to 0 while double would be initialized to 0.0.

11.Which type can fill in the blank?

```
pi = 3.14;
```

A. byte

B. float

C. double

D. short

C. Options A and D are incorrect because byte and short do not store values with deci- mal points. Option B is tempting. However, 3.14 is automatically a double. It requires casting to float or writing 3.14f in order to be assigned to a float. Therefore, Option C is correct.

12.Which of the following is not a valid class declaration?

A. class building {}

B. class Cost$ {}

C. class 5MainSt {}

D. class _Outside {}

C. Class names follow the same requirements as other identifiers. Underscores and dol- lar signs are allowed. Numbers are allowed, but not as the first character of an identifier. Therefore, Option C is correct. Note that class names begin with an uppercase letter by convention, but this is not a requirement.

13.Which is correct about a local variable of type String?

A. It defaults to an empty string.

B. It defaults to null.

C. It does not have a default value.

D. It will not compile without initializing on the declaration line.

C. Local variables do not have a default initialization value. If they are referenced before being set to a value, the code does not compile. Therefore, Option C is correct. If the variable was an instance variable, Option B would be correct. Option D is tricky. A local variable will compile without an initialization if it isn't referenced anywhere or it is assigned a value before it is referenced.

14.Given the following code, fill in the blank to have the code print bounce.

```
public class TennisBall
{
  public TennisBall()
  {
   System.out.println("bounce");
  }
  public static void main(String[] slam)
  {

  }
}
```

A. TennisBall;

B. TennisBall();

C. new TennisBall;

D. new TennisBall();

D. In order to call a constructor, you must use the new keyword. It cannot be called as if it was a normal method. This rules out Options A and B. Further, Option C is incorrect because the parentheses are required.

15.Which of the following correctly assigns animal to both variables?

I. String cat = "animal", dog = "animal";

II. String cat = "animal"; dog = "animal";

```
III.   String cat, dog = "animal";
 IV.   String cat, String dog = "animal";
```

A. I
B. I, II
C. I, III
D. I, II, III, IV

A. Option A (I) correctly assigns the value to both variables. II does not compile as dog does not have a type. Notice the semicolon in that line, which starts a new statement. III compiles but only assigns the value to dog since a declaration only assigns to one variable rather than everything in the declaration. IV does not compile because the type should only be specified once per declaration.

16.Which of the following does not compile?

A. double num = 2.718;
B. double num = 2._718;
C. double num = 2.7_1_8;
D. None of the above; they all compile.

B. Underscores are allowed between any two digits in a numeric literal. Underscores are not allowed adjacent to a decimal point, making Option B the correct answer.

17.Fill in the blank to make the code compile:

```
package animal;
public class Cat
{
public String name;
 public static void main(String[] meow)
 {
  Cat cat = new Cat();
       = "Sadie";
}
}
```

A. cat.name
B. cat-name
C. cat.setName
D. cat[name]

A. Java uses dot notation to reference instance variables in a class, making Option A correct.

18.Which is a valid constructor for this class?

```
public class TennisBall
{
}
```

A. public TennisBall static create() { return new TennisBall(); }
B. public TennisBall static newInstance() { return new TennisBall():}
C. public TennisBall() {}
D. public void TennisBall() {}

C. Options A and B are static methods rather than constructors. Option D is a method that happens to have the same name as the class. It is not a constructor because constructors don’t have return types.

19.What is the output of the following?

```
package beach;
public class Sand
{
public Sand()
  {
   System.out.print("a");
  }
  public void Sand()
  {
   System.out.print("b");
  }
  public void run()
  {
  new Sand();
  Sand();
  }
  public static void main(String... args)
  {
    new Sand().run();
  }
  }
```

A. a
B. ab

C. aab

D. None of the above

C. The main() method calls the constructor which outputs a. Then the main method calls the run() method. The run() method calls the constructor again, which outputs a again. Then the run() method calls the Sand() method, which happens to have the same name as the constructor. This outputs b. Therefore, Option C is correct.