iu
INTERNATIONAL
UNIVERSITY OF
APPLIED SCIENCES

Q&As

Units 9,8,7,6,5,4

2/8/2023

Suhyl Mekkawy

1- What is a new operator ?

-This is an operator that is used to create new objects for a particular class

2-What is the standard constructor?

-This is a special method that can exist once in a class and is used to initialize variables of the attributes of an object inside of a class

3- What are the properties of the standard constructor?

1- Its visibility modifier is always public
2- It doesn't have a return data type
3- It has the same name as the class
4-Defined without the parameter list
5- Exist once inside of a class

4-How do the new operator, object and the standard constructors interact with each other?

The new operator creates an object inside of a class then invokes the standard constructor which initialize the value of an object

5-Standard constructor can exist more than once inside of a class (T/F) (once)

6-Write down the syntax of the standard constructor.

- **Public** (name of the constructor as the name of its class) **();**

7-Why is the visibility modifier of the standard constructor always public?

- It allows objects to be created from any other class via the new operator

8-What differs the standard constructor from the overloaded constructors?

-The standard constructor has no parameter and two tasks instead and can only exist once inside of a class but overloaded constructors can have different sets of parameters and can exist more than once inside of a class

9-What are the 2 tasks assigned to the standard constructor?

1- Adapting the default values for primitive data types
2- Creating non-primitive attributes (objects)

10-What happens to the standard constructor if both of its tasks are not necessary for a particular class?

The definition of the standard constructor is omitted and Empty standard constructor is added by the Java compiler

11-What does Empty standard constructor mean?

It is a constructor added by the compiler in case of the standard constructor is not defined, It doesn't initialize the variables value inside of an object as the standard constructor does

12-What is garbage collection in Java?

It is a concept in java that is used by the java runtime environment to delete unused objects automatically to free up space.

13-What is the difference between Java and other programming languages concerning freeing up space?

Java uses the garbage collection concept to automatically delete unused objects while in other programming language we have to manually write down statements for the deletion which saves more time using Java

14-Can we invoke garbage collector manually in Java and if yes then what statements we use to invoke it?

Yes we can if we need to free up more space by using the statement System.gc()

15-What are overloading standard constructor?

It is a constructor that is used to initialize a variable value of objects that is only determined at runtime and it has a parameter list and used when there are so many attributes

16-What is the use of parameter list in an overloaded constructor?

Determines which attributes can be used to fill the object with information at runtime

17-When can we use an overloaded constructor?

When there are so many attributes in a class that can be difficult to determine whether all needed attributes have actually been set or an attribute is missing ,Thus increase flexibility

18-Overloaded standard constructor and standard constructor can be used in one class (T/F)

19-Why is it important to have overloaded constructors invoke each other ?

1- To avoid double code
2- To improve serviceability
3- To increase flexibility

20-Which statement can we use to invoke other constructors within the overloaded constructors ?

this(),This statement must be in the first position

21-Define copy constructor.

This constructor is used to clone objects whether a deep(reference data type) copy or a shallow copy(primitive data type) are created

22-What is the keyword used to invoke the constructor of the super class?

super(),with or without parameter list

23-What is the exception?

This is a state in which the program is unable to continue to function normally due to error signals

24-What are error signals?

This is a concept that codes exceptions when using values outside the valid value range, Then the calling program must recognize the exception coding in order to respond with an error signal

25-How do the error signals work?
When using a value that is outside the valid value range then the calling program interpret the exception signals and then translate it into a readable error message by the user

26-What is the disadvantage of conventional error handling situations with error signals?

Functions with only one return value and must return error signals and domain values alike to the calling program to be translated, which can result poor software quality and should be avoided

27-What can be done to avoid the disadvantage of conventional error handling situations with signals?

1-There should be a separate channel for error signals return that results in the decoupling of domain return values and error signals

2- Force all calling programs to catch possible exceptions of a called method

28-What is an exception in Java ?

This is a class that forms the core of object oriented exception handling in Java in which subclasses can be used to handle many different exception situation

29-What are the alternatives for handling the exception in Java?

1- The exception is caught within the method
2- The exception is passed to the calling program
3- The exception is caught and is then passed to the calling program with a specific message

30-Mention excerpts of predefined standard exceptions in Java

Predefined standard exceptions in java	
Name	**Description**
ArithmeticExceptions	Division by zero
ArrayIndexOutOfBoundsExeption	Access to a list element that is outside the defined area
NullPointerException	Access to a non instantiated object

31- Define the try/catch block
They are features of the exception handling in oop in critical program instruction in which the try block encloses section code of the exception for the catch block to handle the exception, They are used to separate the domain return value and the error signal

32-What is the keyword "throws" used to?

This keyword indicates in the method signature which exceptions can be thrown by method (enables the creation of an exception)

33-What is the mechanism of throws/Passing exceptions?

By using throws in the method signature, Calling programs can recognize which exceptions can be expected by this method then exception handling can be moved to calling program by try/catch block

34- What is the method that is used to obtain the error message of the exception object?

getMessage()

35-Several catch blocks can be defined for a single try block (T/F)

36-Why does it make sense that several catch blocks can be defined for a single try block ?

When many exceptions are possible in the try block and each of them has to be handled separately by many other catch blocks

37-What happens if the special catch blocks match ?

The code within the special catch block is executed and the catch block handles the exception

38-What happens if none of the special catch blocks match ?

The exception is caught in the general catch block

39-Why is the exception caught in the general catch block when the special catch blocks don't match?

Because the general catch block is used as a fallback mechanism to handle any exceptions that were not explicitly caught by the special catch blocks. It ensures that no exceptions go unhandled and provides a way to handle unexpected or unknown exceptions.

40-Why is it necessary to ensure that only necessary cleanup tasks are performed in exception handling with try/catch blocks?
To avoid such redundant code

To avoid such redundant code

41-What is the major disadvantage of exception handling with try/catch blocks?

When the familiar language constructs complicate the program over time whenever a program logic has to be invoked every time we execute a try block so there will be so many redundant code

42- What is the finally block?

This is a block that uses mandatory clean up statements to clean up the redundant code from the try/catch block whether the exception has been caught or handled or not

43-Define user-defined exceptions?

Exceptions that are created by the user and must be derived from the Exception class

44-Give a keyword defining the user-defined exceptions.

MinimumOrderValueNegativeException()

45-Each exception, regardless of whether it is the standard exception or a user-defined exception, must be derived from the superclass Exception (T/F)

46-What is the relationship between the constructors and exceptions?

1- The constructors can be used to define which messages are created by the exception

47-What would happen if a standard constructor was used to create messages instead of the exception?

a standard error message should be defined.

4**8-What can we do to create an exception with an error message that is not known until runtime?**
We can add another constructor with a parameter that passes the error message that is sent immediately to constructor of the super class

49-What is a gang of four?

A group of four authors who have written frequently cited literature on the subject of the software technology

50-What is implementation?

It is the technical implementation (programming) of the designated program into an executable code that can be determined by interface

51-What is the use of the gang of four concept?

It ensures a clear separation of the specification and the implementation

52-What is the advantage of separating the specification and the implementation ?

1-It allows more flexible software architecture as the implementation can be changed without significant adaptation

2- It increases the usability of classes or packages

53-What is an interface?

It is a number of methods that can be implemented by classes

54-What is(use association)?
This association indicates that a class uses an interface

55-Where is the Information of objects in object-oriented programming generally stored ?

- Inside if the attributes of an object

56-What determines the type of information stored in the attributes of objects ?

- The data type of the attributes

67-What is the result that affects the object when the name of a class is defined as the data type of an attribute?

- All the objects inside this class are assigned as values to this attribute

68-For integers, the data type (int) is usually used; smaller data types like byte or short are only used for programming (microcontrollers).

69-Define primitive data types

They are data types whose values are not objects, And they are a simple data types that are not described by their own class

70- Define the string data type

It is a data type that is used to store strings.It has the properties of primitive data types but it is considered as a class.

70-What are the values of a string data type ?

They are java objects

71-Like a primitive data type, String can be assigned to attributes (and variables). (T/F)

72-What are the keywords in java and description of String as a data type for strings in Java?

- **Keyword**: String
- **Description**:
  1- Used to store strings of any length

2- class string contain methods that are used to processing strings

73-Mention all the primitive data types in java, their keywords, Descriptions and an example on each of them

Primitive data type	Keyword	Description	Example
Logical values	Boolean	stores values of **true or false,** Other values are not allowed	True,false
Character	Char	Stores a **single unicode** character	Any **numerals, letters, symbols and control characters** such as blanks, tabulator and line break. 'A' (= letter A), '2' (= numeral 2), '\n' (= control character line break)
Integer	byte	**Stores 8**-bit value range from -128 : 127	
	short	**Stores 16-**bit value **-32**,768 : **32**,767	
	int	**Stores 32**-bit value **-2**,147,483,648 : **2**,147,483,647	
	long	**Stores 64**-bit value (8 bytes) **–9**,223,372,036,854,775,808 : **9**,223,372,036,854,775,808	

Floating point numbers	Float	**Stores 32**-bit value range from **1.4**0239846*10–45 (-30.9760154) to **3.4**0282347*1038 (3532.130762) Size = (4 bytes)	**1.87236f** (simple number with decimal point and appended "f") **–3.938e12f** **(**–3.938*1012 with appended "f"**)**
	Double	**64-**bit value (8 bytes) **4**.94065645841246544*10–324 (-274.5934354) To **1**.79769131486231570*10308 (18530.60207)	**1.87236d** (Simple number with decimal point and appended "d"); **–3.938e120d** (– 3.938*10120 with appended "d")

74-What is the use of variables for java?

In Java,Variables allows concrete values to be saved in the application memory

75-What is required to be defined to declare a variable in java?

The name and the data type of the variable

76-The declaration of a variable (including all subsequent assignments, method invocations, and calculations) ends with a (; *semicolon*).

77-Where is the only place that a variable can be saved?

Inside a method body

78-Can we assign any values to a variable before its declaration ?
No

79-How the assignment of the variable using the equal sign is expressed?

The data type and the name of the variable is on the left side of the "=" on the right side of the "=" is the value assigned to the variable (ex: int x=4;)

80-What is the general use of operators

Used to calculate and change values that are stored in variables and attributes

81-What is the data type of the operation result of the operator is determined by?

-It is determined by the operand with the largest value range

82-What is the use of logical operators?

- They return the values of true or false for expressions, Therefore used for managing control structures

83-The result data type of logical and relational operators is always (boolean.)

84-What is the use of Relational operators

- Compare expressions against each other, And used for managing control structures

85-What are test equality operators?

- These are relational operators that are used as the test for equality (==) and the test for inequality (!=) of the expressions, as the data type of the operands determine whether values or references are compared

87-If both operands are primitive data types, then the values of the operands are compared indirectly.(T/F)

88-The operator = = does not compare the actual contents of the objects being compared. Instead, the function only compares (whether both operands refer to the same object or not**)**

89-What is string concatenation ?

- This is a function/operation of the data type string which creates new string from multiple existing strings through simply stringing/joining them together

90- Mention the operator,description,data type and example of string concatenation

Operations	Description	Operator	Data type	Example
String joining **(String concatenation)**	creates new string from multiple existing strings through simply stringing them together	+	string	**String s1, s2, s3, s4;** **s1 = "Hello";** **s2 = " ";** **s3 = "World!";** **s4 = s1 + s2 + s3;**

91- Mention types of operators in Java

1- Arithmetic operators
2- Logical operators
3- Relational operators

92-Mention the operator,description,data type and example of the arithmetic operators

Operations	Description	Op	Data type	Example
Increment	Increases a variable by the value 1	++	-Integers -Floating point numbers	**Int a=3;** **Int b= 4;** **Int c= b++;** **Result: c=5**
Arithmetic addition	Adding the values of both operands The result type of the calculation corresponds to that of the operand with the largest value range (e.g., int+int=int, int+long=long;)	+	-Integers -Floating point numbers	**Int a1= 3;** **Int a2= 4;** **int3= a1+a2;** **result=7**
Arithmetic subtraction	Subtract the values of both operands	-	-Integers -Floating point numbers	**Int a1= 3;** **Int a2= 4;** **int3= a1-a2;** **result= -1**

Arithmetic multiplication	Multiplies both operands	*****	-Integers -Floating point numbers	**Int a1=3;** **Int a2=4;** **float a3=5.46f;** **Int a4= a1*a2;** **Result=12** **float a5= a4*a3;** **result= 17.46f**
Arithmetic division	Calculates quotients of dividend and divisor (divides both operands)	**/**	**Integer :** if both operands are integers, the part after the decimal point is truncated **Floating point numbers** : if at least one of the operands is floating point number, the result is a fpn and is not rounded	**Int a1=3;** **Int a2=4;** **Flaot a3=4.56f;** **Int a4=a1/a2;** **result = 0** **Float a5= a1/a3;** **result= 0.65f**
Remainder **(remainder operator/ modulus)** **(modulo operator)**	Calculates the remainder of the arithmetic **division**	**%**	-Integers -Floating point numbers	**Int a1=4;** **Int a2=5;** **Int a3= a1%a2;** **Result 20**

93-if both operands are integers, the part after the decimal point is truncated (t/f)

94-if at least of of the operands is floating point number, the result is a floating point number and is not rounded (t/f)

95--Mention the operator,description,data type and example of the logical operators

Operations	Description	Operator	Data type	Example(int a=2;b=3;)
Logical NOT	It reverses the meaning of the operand	**!**	boolean	**Boolean a=true;** **boolean=!a;**

Logical AND	Returns True if both operands are true	**&&**	**boolean**	**Boolean a=true;** **Boolean b=true;** **Boolean c=a&&b;**
Logical OR	returns true if either of the operands is true - It evaluates the second operand only if the first operand is false. - It is a short-circuiting operator, meaning that if the first operand is true, it does not evaluate the second operand.	||	**boolean**	**Boolean a=true;** **Boolean b=false;** **Boolean c=a||b;**
Exclusive OR	Returns true only if exactly one of the operands is true. - It evaluates both operands regardless of their values. - It does not short-circuit and always evaluates both operands.	^	**boolean**	**Boolean a=false;** **Boolean b=true;** **Boolean c=a^b;**

96--Mention the operator,description,data type and example of the relational operators

Operations	Description	Operator	Data type	Example(int a=2;b=3;)
Equality primitive data types	returns **true** if the values of the operands are equal.	= =	Primitive data type (boolean)	**Int a1=3;** **Int a2=3;** **Boolean a3=a1==a2;**
Equality Reference data types	Returns **true** if the same object is referenced in both operands.	= =	**Reference data type**	**Customer customer1, customer2;** **boolean e2;** **customer1 = new**

				Customer(); **customer2 = customer1;** **e2 = customer1 == customer2;**
Non Equality primitive data types	returns **true** if the values of the operands are not equal.	**!=**	Primitive data types (boolean)	**int z3, z4;** **boolean e3;** **z3 = 4;** **z4 = 3;** **e3 = z3 != z4;**
Non Equality Reference data types	Returns **true** if the two operands contain different references	**!=**	**Reference data type**	**Customer customer3, customer4;** **boolean e4;** **customer3 = new Customer();** **Customer4 = new Customer();** **e4 = customer3!= customer4;**
Less than	Returns true if the value of the operand on the left is less than the value of the operand	**<**	Integers and floating point numbers	**int z5, z6;** **boolean e5;** **z5 = 4;** **z6 = 5;** **e5 = z5 < z6;**
Less than or equal to	Returns true if the value of the operand on the left is less than or equal to the value of the operand on the	**<=**	Integers and floating point numbers	**int z5, z6;** **boolean e5;** **z5 = 4;** **z6 = 5;** **e5 = z5 <= z6;**

	right.			
Greater than and greater than or equal to	The opposite description of less and less than operators above	**>** **>=**		
Type comparison	returns true if the data type of the operand on the left is the same as the	**instanceof**	**Reference data type**	**Customer k1;** **boolean e7;** **k1 = new Customer();** **e7 = k1 instance of Customer;**

97-What are control structures and what is the use of them?
These are elements of a programming language for controlled repeated execution of statements whether conditional or nested execution inside the method body

98-The getter methods for attributes of the type boolean begin with "is**" instead of "get."**

99-What is the conditional branch ?

This is a control structure that is used to execute conditional statements inside the method body

100-What is the main property of a condition?

It must be an expression that can be evaluated to be true or false.

101-Why is the else part in conditional branches optional ?

Because the Statement1 is only executed if the condition is true. In this case, Statement2 is not executed.And is only executed if the condition is false

102-What is the expanded if-else branch?

This is a control structure that is used to execute multiple conditions statements

103-What is a switch in java?

It is a control structure for complex branches in java and it can be replicated by using expanded if-else branch

104-What are loops?

They are a control structure used for executing statements repeatedly.

105-What is the number of iterations of loops determined by ?

It is determined by the satisfaction of loop conditions

106-What is a while loop ?

This is a pre checked loop, in which the conditions are tested before the execution of the statements

107-Mention the mechanism of the while loop./Why is the while loop called a pre-checked loop ?

The conditions of the loops are tested and if the result is true then the statements of the loop are executed otherwise the statements are skipped, After the execution of the statements the condition is evaluated again and if it is evaluated to false then the loop ends, But if it is evaluated to true then the loop is repeated again.

108-What is the truth test of the condition based on in a while loop?

The truth of the tested condition is based on the current value of index and changes based on the index during processing of the loop

109-What is a do-while loop ?

The is a post checked loop, In which the conditions are tested after the execution of at least statement

110-Mention the mechanism of do-while loop./Why is the do-while loop called a post-checked loop ?

The conditions of the loops are tested after the execution of at least one of its statements, if the result is true then the loop is repeated otherwise the statements are skipped,

111-What is a For loop?

It is a pre-checked loop which test the condition of the loop before execution of its statements, But it differs from the while loop that the head, in addition to the condition, also contains the statements for

initialization of the counter variables and their incrementation.

112-Control structures can themselves contain control structures in their statement blocks.Therefore, loops can also be nested with each other. (T/F)

113-What are the most important of the Java-supported control structures?

1-Conditional branches(if-else)
2- Loops

114-Mention the three types of loops.

1- While loop
2- Do while loop
3- For loop

115-What does the number of iterations inside of the loop depends on ?

- It is determined by the interaction between:

- The initial value of the index
- The statements that changes the value of the index
- The loop condition

116-The flow schema of a for loop can be described as follows;

1- Execute the statements for initialization
2- Test the condition
3- Evaluate the condition
If true : execute all the statements if the for loop
If false : abort,no execution
4- Execute the statements in the loop increment
5- Continue with the second iteration

117-What are packages?

This is an element that is used in structuring classes to which similar classes that are closely dependent on each other are assigned as well as it can contain other packages

118-(Directories) provide means for structuring files but don't save information

119-What is the difference between directories and files?

Every directory can contain subdirectories but a file can not contain additional directories or files

120-What is the effect of packages on their assigned classes

The storage location of the class is affected by the package assignment and on the access authorization to methods of other classes

121-The first line of the source text in a Java class always contains the package in which the class is located (T/F)

122-Mention how the relationship between the package,directory and file react with each other

A directory is created for each package of a program. The file with the Java class must reside in the directory of the package.

123-What are visibility modifiers?

They are elements of Java that are used to define the visibility of classes atrtributes and methods for other classes

124- Mention the types of visibility modifiers, their description and where it can be used

Visibility modifier	Description	Can be used with	example
public	The element is visible to all classes in the program.	**Classes** **Attributes** **Methods**	-public class Class{..} -Public int attribute {..} -Public string method(){..}
None	The element is only visible in the same package	**Classes** **attributes** **methods**	class Class{..} string attribute {..} string method() {..}
protected	The element is only visible in the same package or derived	**attributes** **methods**	Protected string attribute {..} Protected string method() {..}

	classes.		
Private	The element is only visible for elements of the same class.	**attributes methods**	Private string attribute {..} Private string method() {..}

125-What is a superclass/Base class?

It is a class that contains all the mutual attributes of all the associated classes by 'Is a" association.

126-What is the UML class diagram notation for the superclass?

A line between the derived classes and the super class that ends with a hollow arrow head in the super class

127-The "is a" relationship is an important concept in object orientation and is classified as an inheritance relationship

128-What is a subclass/derived class?

It is a class that is derived from the superclass and associated to the superclass by the inheritance relationship by which it inherits the superclass attributes and methods but it has an additional attributes for specification of its description

129-What is the general use of inheritance relationships ?

It allows the associated subclasses to inherit the mutual attributes and methods of the superclass but at the same time each subclass has its own specific attributes and methods which specify it from other subclasses thus allows information to be conveyed either generally or specifically while avoiding redundancy

130- How do you explain the phrase 'Inheritance relationship is transitive' ?

This means the attributes of an ancestor class are passed down to all child classes.

131-What is being shared in an inheritance relationship between classes?

Inheriting attributes and methods from the superclass not only form the structure of a subclass but its properties and behavior

132- How can objects of a subclass be used in association with the super class?

The objects in the subclass inherits the methods and attributes of the superclass and can be addressed in the same way with the inherited association

133- Any changes in the super class like an attribute being added or removed the changes will be inherited by the subclasses (T/F)

134- Why are the attributes and methods of the superclass not repeated ?

1-To avoid redundancy
2- To improve the clarity of the system design in the model

135-From which superclass would a subclass inherit the attributes and methods from in case of multiple superclasses ?

From the union of the attributes and methods of all the superclasses

136-What is extension in java?

It is a concept that is used by the keyword extends in the declaration of the subclass to add specialization/derivation for the superclass

137- In Java the concepts of specialization and derivation are called “Extension”

138-How is the keyword extends assigned in Java/How inheritance is associated in Java?

The keyword extends is added to the declaration of the subclass after the subclass name then we mention the superclass name from which this subclass is derived from.

139-All the attributes and methods that are considered as private in the superclass shouldn’t be inherited by the subclass (T/F)

134-What is assignment compatibility?

This means that a variable of the type of a superclass can also be assigned by subclass objects.

135- Explain in an example how the data type of the variable affects the subclass process.

If a variable whose reference data type is the superclass that is used by the subclass objects then the methods used must be defined in the superclass

136-Although subclasses inherit the attributes and methods of the superclass, they are strictly tied to the implementation.(T/F)

137-What is overriding?

It is the process of implementing an inherited method from the superclass to the subclass

138-How methods are overriding in a subclass?

By creating a method in the subclass with the same signature as it is in the superclass and implementing a new method body

139-What is the keyword that is used to access existing implementations(methods and attributes) in the super class?

Super

140-How does the keyword super work?

Because each subclass inherits from one superclass, super adds a clear definition of what class is intended, super goes up one level in the inheritance hierarchy and select the implementation there

141-What is the abstract class?

This is a class in which all the common features are grouped together and any instances can't be created within

142-How should the abstract class be designated In UML?

the abstract class must be designated with the word abstract inside of curly braces in UML

143-What is the keyword used when assigning an abstract class?

abstract, After the visibility modifier in the class declaration

144-What happens to the class that was assigned to be abstract?

No instances can be created from it, Otherwise they can be created like normal classes

145-What are the properties of the abstract class?

1- It contains the common features and functions of the derived classes
2- Attributes and methods can be inherited by the derived classes from it
3- Any Instances can't be created from it once the keyword instance is assigned to this class

146-What is the abstract method?

-This method has a signature but not a method body, It defined the functionality and behavior of the objects that must be implemented in subclasses

147-If a class contains at least one abstract method, The class must be declared as abstract,Otherwise the class is incomplete (T/F)

148-Abstract methods must be implemented in concrete subclasses by overriding after it has been declared by an abstract class (T/F)

149-What is the only possibility for not implementing an abstract method inside of a direct subclass?

By assigning it as an abstract again in the subclass,This results in this subclass becoming an abstract class and the responsibility for implementation being delegated to other subclasses.

150-How can we declare a variable in an abstract class?

We can declare variables in an abstract class only when we assign the value of the variable as objects from a derived class

151-What is polymorphism?

This means that objects of different classes can be assigned to the same variable when the classes have an inheritance relationship with each other.

152-What element determines which method to be invoked?

The type of the variable

153-What determines which implementation of the method inside of the class?

The instances inside of a class

154-What are the functions of the concept of polymorphism?

1- It allows objects of different classes to be treated as objects of a common superclass.

2- It allows a single variable of a superclass to be assigned to objects of different subclasses classes

3- It allows method implementation by objects of different classes without the need to know which subclasses these objects belong to

155- What are class variables(Static attributes)?
These attributes are the same for all instances of a class and are addressed through the class name, Since they are not bound to an object but they are already available from the time the class exists.

156-Mention the static attributes/class variables properties.

1- They are the same for all instances of a class and apply to all of them
2- Their value is shared by all the instances of a class
3- They are declared using the keyword static
4- They don't bound to objects in the class

157-What will happen to all the objects of the classes and subclasses if we change the value of the static attributes?

When the value of the static attribute is changed, then it changes for all the objects of the class and the subclasses.

157-What are the properties of a static method?

1- They don't bound to objects of the class
2- They are invoked through the class
3- They are invoked using the keyword static

171-Why the instances of the static method body can't be accessed?

Because static methods can be invoked without the existence of objects of the particular class,(Doesn’t bound to an object)

172-Static attributes are often found in the Java class library with the definition of ”Constants” For example, the number Pi and Euler's number e are defined as static attributes of the class “Math”

173-The output of the methods is strictly determined by the values of the **(Parameter list)** alone.