Programs on pre-defined function interface methd

Q1.What is a Supplier in Java?

a) It's a class used for creating instances of objects.

b) It's a functional interface representing a supplier of results.

c) It's a utility class for handling input/output operations.

d) It's a keyword used to define package-private access.

Answer: b) It's a functional interface representing a supplier of results.

Q2.Which method is defined in the Supplier interface?

a) accept()

b) supply()

c) get()

d) provide()

Answer: c) get()

Q3.Which package does the Supplier interface belong to?

a) java.util.function

b) java.util.supplier

c) java.util

d) java.function

Answer: a) java.util.function

Q4.Which of the following best describes the purpose of Supplier in Java?

a) It represents a function that takes an argument and produces a result.

b) It represents a function that produces a result without taking any input.

c) It represents a function that takes multiple arguments and produces a result.

d) It represents a function that modifies the state of an object.

Answer: b) It represents a function that produces a result without taking any input.

Q5.Which of the following is a correct way to use a Supplier in Java?

a) Supplier<String> s = new Supplier<>("example");

b) Supplier<String> s = () -> "example";

c) Supplier<String> s = Supplier.of("example");

d) Supplier<String> s = Supplier.get("example");

Answer: b) Supplier<String> s = () -> "example";

Q6.Which of the following functional interfaces is NOT a part of the java.util.function package?

a) Consumer

b) Supplier

c) Executor

d) Predicate

Answer: c) Executor

Q7.What is the return type of the get() method in the Supplier interface?

a) void

b) T

c) Object

d) Supplier

Answer: b) T

Q8.Which of the following is a valid lambda expression representing a Supplier

that generates a random integer between 1 and 100 (inclusive)?

a) () -> Math.random() \* 100

b) () -> (int) (Math.random() \* 100)

c) (int) -> (int) (Math.random() \* 100)

d) () -> new Random().nextInt(100) + 1

Answer: b) () -> (int) (Math.random() \* 100)

Q9.Which method of the Stream interface utilizes a Supplier to generate

elements for the stream?

a) forEach()

b) filter()

c) map()

d) generate()

Answer: d) generate()

Q10.What is the purpose of the Function interface in Java?

a) It represents a function that produces a result without taking any input.

b) It represents a function that takes an argument and produces a result.

c) It represents a function that modifies the state of an object.

d) It represents a function that handles exceptions during input/output operations.

Answer: b) It represents a function that takes an argument and produces a result.

Q11.Which method is defined in the Function interface?

a) apply()

b) get()

c) call()

d) execute()

Answer: a) apply()

Q12.Which package does the Function interface belong to?

a) java.util.function

b) java.util.functionality

c) java.function

d) java.util

Answer: a) java.util.function

Q13.Which of the following represents a correct lambda expression for a Function that converts a String to its length?

a) (String s) -> s.length()

b) (String s) -> return s.length()

c) (String s) { return s.length(); }

d) (s) -> s.length()

Answer: a) (String s) -> s.length()

Q14.What is the return type of the apply() method in the Function interface?

a) void

b) T

c) Object

d) R

Answer: d) R

Q15.Which of the following functional interfaces represents a function that takes no arguments and produces a result?

a) Consumer

b) Supplier

c) Predicate

d) Function

Answer: b) Supplier

Q16.What does the compose() method in the Function interface do?

a) It applies the given function to the result of this function.

b) It applies the given function before applying this function.

c) It combines the given function with this function using logical AND.

d) It combines the given function with this function using logical OR.

Answer: b) It applies the given function before applying this function.

Q17.What does the andThen() method in the Function interface do?

a) It applies the given function to the result of this function.

b) It applies the given function before applying this function.

c) It combines the given function with this function using logical AND.

d) It combines the given function with this function using logical OR.

Answer: a) It applies the given function to the result of this function.

Q18.Which of the following is a correct way to use a Function in Java?

a) Function<String, Integer> func = (int x) -> x.length();

b) Function<Integer, String> func = (int x) -> x.toString();

c) Function<Double, Integer> func = (double d) -> d \* 2;

d) Function<Integer, Integer> func = () -> x \* x;

Answer: b) Function<Integer, String> func = (int x) -> x.toString();

Q19.What is the purpose of the Consumer interface in Java?

a) It represents a function that produces a result without taking any input.

b) It represents a function that takes an argument and produces a result.

c) It represents a function that modifies the state of an object.

d) It represents a function that handles exceptions during input/output operations.

Answer: c) It represents a function that modifies the state of an object.

Q20.Which method is defined in the Consumer interface?

a) accept()

b) apply()

c) consume()

d) get()

Answer: a) accept()

Q21.Which package does the Consumer interface belong to?

a) java.util.function

b) java.util.consumer

c) java.consumer

d) java.util

Answer: a) java.util.function

Q22.Which of the following represents a correct lambda expression for a Consumer that prints the elements of a list?

a) (List<T> list) -> System.out.println(list)

b) (T t) -> System.out.println(t)

c) (t) -> System.out.println(t)

d) () -> System.out.println(list)

Answer: c) (t) -> System.out.println(t)

Q23.What is the return type of the accept() method in the Consumer interface?

a) void

b) T

c) Object

d) R

Answer: a) void

Q24.Which of the following functional interfaces represents a function

that takes no arguments and produces a result?

a) Consumer

b) Supplier

c) Predicate

d) Function

Answer: b) Supplier

Q24.What does the andThen() method in the Consumer interface do?

a) It applies the given function to the result of this function.

b) It applies the given function before applying this function.

c) It combines the given function with this function using logical AND.

d) It combines the given function with this function using logical OR.

Answer: b) It applies the given function before applying this function.

Q25.What is the purpose of the accept() method in the Consumer interface?

a) It returns a result.

b) It modifies the state of an object.

c) It performs a computation.

d) It handles exceptions during input/output operations.

Answer: b) It modifies the state of an object.

Q26.Which of the following is a correct way to use a Consumer in Java?

a) Consumer<String> consumer = (String s) -> s.toUpperCase();

b) Consumer<Integer> consumer = (int x) -> System.out.println(x);

c) Consumer<Double> consumer = (double d) -> d \* 2;

d) Consumer<String> consumer = () -> System.out.println("Hello");

Answer: b) Consumer<Integer> consumer = (int x) -> System.out.println(x);

=-----------------------------------------------------------------------------------------------------------------------------------

Programming question:

Q1.Write a Java program that demonstrates the usage of Supplier to generate

a random number.

**public** **class** MyProgram

{

**public** **static** **void** main(String[] args)

{

Supplier s=() ->

{

**return** Math.*round*(Math.*random*()\*99999);

};

System.***out***.println("Random number :"+s.get());

}

}

Q2.Write a Java program that utilizes a Supplier to generate a sequence of

Fibonacci numbers. The program should prompt the user to enter the number

of Fibonacci numbers they want to generate, and then print out the sequence.

**import** java.util.Scanner;

**import** java.util.function.Supplier;

**public** **class** MyProgram

{

**public** **static** **void** main(String[] args)

{

Supplier fabon=() ->

{

System.***out***.print("Enter number of Fibonacci numbers you want to generate :");

Scanner sc=**new** Scanner(System.***in***);

**int** size =sc.nextInt();

**int** a[]=**new** **int**[size];

**if**(size==1)a[0]=0;

**else** **if**(size==2)

{

a[0]=0;

a[1]=1;

}

**else**

{ // f s

**int** f=0,s=1,t=0; // f s

a[0]=f;

**for**(**int** i=1;i<size;i++)// 0 1 1 2 3 4

{

a[i]=s;

t=s;

s=f+s;

f=t;

}

}

**return** a;

};

**int** []arr=(**int**[]) fabon.get();

System.***out***.print("Fabonic sequence :");

**for**(**int** x : arr)

{

System.***out***.print(x+" ");

}

}

}

Q3.Write a Java program that utilizes a Consumer to modify a list of strings.

The program should prompt the user to enter a list of strings separated by

spaces, then convert each string to uppercase using a Consumer, and finally

print out the modified list of strings.

**import** java.util.Scanner;

**import** java.util.function.Consumer;

**public** **class** MyProgram

{

**public** **static** **void** main(String[] args)

{

System.***out***.print("Enter size of list :");

Scanner sc=**new** Scanner(System.***in***);

**int** size=sc.nextInt();

System.***out***.print("Enter "+size+" Strings separated by spaces:");

String [] li=**new** String [size];

**for**(**int** i=0;i<size;i++)

{

li[i]=sc.next();

}

Consumer<String []> c=(String [] t) ->

{

System.***out***.print("list in upper case :");

**for**(**int** i=0;i<t.length;i++)

{

System.***out***.print(li[i].toUpperCase()+ " ");

}

};

c.accept(li);

}

}