Q1) Create a Pangram checker in java (A pangram is a sentence containing every letter in the English Alphabet (a-z).)

Pangram example: The quick brown fox jumps over the lazy dog

Answer=<https://github.com/nskadam02/JavaExercise.git>

Q2) Write a class called MyRegex which will contain a string pattern. You need to write a regular expression and assign it to the pattern such that it can be used to validate an IP address. Use the following definition of an IP address:

IP address is a string in the form "A.B.C.D", where the value of A, B, C, and D may range from 0 to 255. Leading zeros are allowed. The length of A, B, C, or D can't be greater than 3.

Some valid IP address:

000.12.12.034

121.234.12.12

23.45.12.56

Some invalid IP address:

000.12.234.23.23

666.666.23.23

.213.123.23.32

23.45.22.32.

I.Am.not.an.ip

Answer:

class MyRegex  
{  
 String pattern =  
 "^([01]?\\d\\d?|2[0-4]\\d|25[0-5])\\." +  
 "([01]?\\d\\d?|2[0-4]\\d|25[0-5])\\." +  
 "([01]?\\d\\d?|2[0-4]\\d|25[0-5])\\." +  
 "([01]?\\d\\d?|2[0-4]\\d|25[0-5])$";  
  
}

Q3) Write three Java lambda expressions

1. One that returns true if a number passed to it is odd and false if it’s even.

answer:

(int no)-> no%2 !=0 ? true: false;

1. One that returns true if a number is prime and false if it’s not.

Answer:

static boolean checkPrime(int n)

{

if (n <= 1)

{

return false;

}

for (int i = 2; i < n; i++)

{

if (n % i == 0){

return false;

}

return true;

}

(int no) -> checkPrime(no) == true ? true : false;

1. One that returns true if the parameter is a palindrome and false if it’s not.

static boolean checkPalindrome(String str)

{

int i = 0;

j = str.length() - 1;

while (i < j)

{

if (str.charAt(i) != str.charAt(j))

{

return false;

}

i++;

j--;

}

return true;

}

(String s) -> checkPalindrome(s) == true ? true : false;

Q4) Is the below statement correct?

ArrayList<int> intArrayList = new ArrayList<int>();

Write the correct way to create an ArrayList of primitive types (int and double).

Answer: Right way to create is:

1)for integer: ArrayList<Integer> arrlist=new ArrayList<Integer>();

2)for Double:ArrayList<Double> aarlist=new Arraylist<Double>();

Q5) What is the difference between Polymorphism, Overloading, and Overriding?

Answer:Polymorphism is the ability of method to behave differently at different places .Single name multiple behaviour.

It is acheived by overloading (compile time polymorphism) and overriding(runtime polymorphism)

1)to acheive overloading we require only one class amd to acheive overrriding we reuire two claasses.

2)overloading implements compile time polymorphism whereas overriding implements runtime polymorphism

3)In overloading method names are same but parameters are different but in overriding method names and arguments both are same

Q6) Why Java doesn't support Multiple Inheritance in Java?

Answer: Java doesnt support multiple inheritance to avoid ambiguity.For example there are three classess A,B and C where B and C Contains same method called demo().Class A extends Class B and C. So while compilation compiler cannot decide which demo method should inherit in Class A.To avoid this ambiguity there is no Multiple inheritance in Java.But java can implement one or more interface to come over this mutliple inheritance problem.

Q7) How can you make a class static in Java?

Answer:We can not make top level class static but we can make nested class static by writing static keyword before class declaration

for eg. Static class demo