Chapter 3

OOPs … JAVA

What is Static in Java?

* Static is the keyword in java used mainly for the memory management and used to create methods and fields that belong to the class rather than to the instance of the class.
* In java for using the variables and methods we create in the class, we need to create the object of the class and call the methods or variables.
* In Java objects are stored in the heap memory and static objects in the stack memory and when we create the object, the static methods or variables will not be part of this object of class and instead they are stored in the common memory of java so whenever we call these static objects, they are loaded without the need of class object.
* So, whenever we want to use the static objects, we need not to create object of the class and we can call them directly by name or using class name.

What can be made static in java?

* Variables, methods, blocks, nested classes.

What are the advantages of using static?

* As they are loaded along with the class and no need to create the object to call them, the memory usage is reduced.
* Also, static fields can't be overridden so they add security.

Can the top-level class be made static?

No, the top-level class in java can never be static because static keyword is meant for providing memory and executing the logic without creating objects, a class doesn't have a value logic directly. Instead, static can be added for inner classes.

Can interface have static classes?

For nested classes we can have static in interface

Can interface have static methods in java?

* Interface in java have abstract methods and fields which are final and static and as the static methods will be loaded in the memory along with the class, they can't be static
* But from java 8, we are allowed to have static methods in interface with body and for using them we need to call them using the name of the interface same as static methods of class.

Can abstract class be static?

* No because abstract class contains abstract methods that should be implemented in derived classes.
* Static keyword in class means all the methods in the class are static nas well, but static methods can't be inherited or overridden so they can't be abstract.

Can a constructor be static?

* No, we know that static belongs to class rather than object of the class and constructor is called when an object is created for the class, so there is no use of static constructor.
* Another thing is that if we will declare static constructor then we can't access/ call the constructor from subclasses as static is allowed with in the class and not to the subclasses

Can static methods be overridden?

No static methods can't be overridden as they are class methods and hence access to them is always resolved during the compile time rather than run time and overriding is happening at run time. When we override a method, we won't get any compile time error, but it won't work in the same way as for non-static methods so method will not be overridden and will work according to the parent class.

Can static methods be overloaded?

Yes, static methods can be overloaded

Can abstract class have static methods?

Yes, abstract class can have static methods, but can't create an abstract static method. So, it will non abstract static method. Reason is that we can’t create object instance to access a static method. so, we need method to be defined with a certain functionality.

Why abstract and static not used together?

Static belongs to the class and when we use abstract, its implementation is provided in other class. So, both classes can't be used together.

Can I have static methods in interface?

From java 8 we can have the static methods in interface along with default methods. But these methods need to be implemented rather than defined in the interface.

Can we have only static block in a class without main method?

Till java 6 it is possible, from java 7 main method is mandatory and without main method it will give the compilation error "Error: main method not found in the class"

Can we reference non static method from static context?

No static methods can be referenced from static context only. Accessing non static from static context give compilation error (can't make static reference to non-static)

When is the static executed?

when the java code is compiled, its bit code is generated, and bit code is used by the class loader to load into the previous memory in the runtime so static is always executed at a class loading.

What is the order of execution for static block, instance block and constructor?

static block -> instance block ->constructor

In which order the static is executed in a program

In the order in which they are written from top to bottom, if we have concept of inheritance involved the parent is loaded first followed by the child

Where is the static context stored?

In java with respect to memory management we have five components - stack, method area, heap, pc register, native area and static is stored in method area (meta space from java 7) and objects are stored in heap.

What are Examples of static usage?

Collections class/ math utility class in JDK, all the methods inside them are static, so to use we can call them by class name and apart from that string utils class is also static.

Do we need to initialize the static variables?

No need, if we don’t initialize them, default values will be assigned to them

What is Constructor in Java?

* Constructor is a block of code, or a special method used to initialize the object, it is called when the instance of the class is created.
* Every time an object is created using the new () keyword, at least one constructor is called.
* when we create an object in java for class we do as classA classa = new classA();
* Here classA() is the constructor which is default - so this is the special method which has name same as class name with no return type and it is used to initialize or assign values to the variables of the objects.
* So, when we create the object for the class, default constructor is called. even we can write the constructor in the class but java by default provides us the default constructor and if we don't mention constructor this default constructor will be called.

What are different types of constructors?

No-arg constructor and parameterized (private, copy) constructor.

Does constructor have return type?

No constructor will not have return type and by default it returns the current instance of the class.

What is the purpose of the default constructor?

Default Constructor provides default values to the object like 0, null etc depending on the type.

What is the use of parameterized constructor?

Used to provide different values to distinct objects. We can also provide the same values.

Can constructor perform other tasks instead of initialization?

* It can also be used for object creation, starting a thread, calling a method, etc.
* You can perform any operation in the constructor as you perform in the method.

Is there Constructor class in Java? – yes

What is the purpose of Constructor class?

It can be used to get the internal information of a constructor in the class. It is found in the java.lang.reflect package.

What are the different modifiers that can be used with constructors?

Constructor should have only public, private, protected keyword as modifier, it can’t be static or final.

Can we overload the constructor?

Yes, we can overload the constructor, by creating the parameterized constructor

Can we override the constructor?

No, as we have different superclass and subclass names, and the constructor’s name always will be same as class name. we can't override the constructor

What is constructor chaining?

* It is the process of calling one constructor from another constructor with respect to current object.
* Constructor chaining occurs through inheritance and a sub class constructor’s task is to call super class’s constructor first.
* This ensures that creation of sub class’s object starts with the initialization of the data members of the super class.
* There could be any numbers of classes in inheritance chain. Every constructor calls up the chain till class at the top is reached.

How is constructor chaining implemented?

* within same class: It can be done using this() keyword for constructors in the same class.
* from base class: by using super keyword to call constructor from the base class.

Why do we need constructor chaining?

This process is used when we want to perform multiple tasks in a single constructor rather than creating a code for each task in a single constructor, we create a separate constructor for each task and make them chain which makes the program more readable.

What is alternative of constructor chaining?

* When we want certain common resources to be executed with every constructor, we can put the code in the init block.
* Init block is always executed before any constructor, whenever a constructor is used for creating a new object.

What is copy constructor?

* There is no copy constructor in java, however we can copy the values from one object to another like copy constructor in C++.
* There are many ways to copy values of one object into another in java.
* By constructor, by assigning the values of one object into another, by clone() method of object class.
* So, copy constructor is special type of constructor that is used to create a new object using the existing object of a class that we have created previously.
* It creates a new object by initializing the object with the instance of the same class.
* The Java Copy Constructor provides a copy of the specified object by taking the argument as the existing object of the same class.

How to create copy constructor?

* To create a copy constructor in Java, we need to first declare a constructor that takes an object of the same type as a parameter.
* After declaring a copy constructor, we need to copy each field of the input object of the class into the new object.

Text

Description automatically generated

What is the use of copy constructor?

* The Copy constructor is easier to use when our class contains a complex object with several parameters.
* Whenever we want to add any field to our class, then we can do so just by changing the input to the constructor.
* there is no need for any typecasting.
* Copy Constructors allow us to change the fields declared as final.
* Using a copy constructor, we can have complete control over object creation.

What is object cloning?

* It means creation of an exact copy of an object, it creates a new instance of the class of the current object and initializes all its fields with the exact values.
* This can be achieved by copy constructor, clone() method.

What is the use of Private constructor?

* A private constructor in Java ensures that only one object is created at a time.
* It restricts the class instances within the declared class so that no class instance can be created outside the declared class.
* Private constructors are used, when we want to make the class as singleton and more secure and restrict from creating more than one instance of the class object.

What is Singleton class?

* We can create only one instance of the class that means classaAobj1= new classA, and we can't create classA obj2= new classA
* Similar to static fields, the instance fields of the class will occur only for a single time.

How can we create a singleton class?

We can achieve this by using private constructor for the class, and creating the object of the class with static keyword in the class and creating the static method in the class which will return the object of the class

Text

Description automatically generated

What is the use of singleton class?

* It restricts the limit of number of object creations to only one which ensures there is access control to resources like db connection etc.
* Memory space wastage doesn't occur with the use of singleton class because it restricts the instance creation.
* Multi-threaded and db applications mostly make use of singleton class for catching, logging, thread pooling, configuration setting etc.

Can the construction have the return type added and how does that work?

Construction can't have a return type and it is what it differentiates from the method.

Can a constructor have a final keyword?

Constructors can't be inherited, they are not subject to hiding or overriding, so there is no change of modifying and so sense of restricting the modification where there is no chance of modifying. And java constructors are internally final.

Can a method have name same as class name?

Yes, we can have method with the name same as class name.

Why main method needs to be static?

For static methods we need not to create objects and main method is the first point of execution so the compiler looks for the main method and we can't have any object before this that means we can't call main method through the object as it is first line. So, to not allow main method to create object we make them static

What happens when we remove static for main method?

It works fine for the compilation but while run time it can't find the method as it looks with the pattern of static void main and string[] as arguments so it throws error.

Can we overload the main method?

Yes we can overload the main method but while running it considers only the method which as string[] as the arguments and executes that first

Can we override the main method?

No as the main method is static it can't be overridden.

What is the first argument of the String array in main() method?

Empty, it does not have any element.

What is Concrete method?

The method which as its own implementation is called concrete method

What is abstract method?

The method which doesn't have its own implementation is abstract method.

What is Abstract class?

The class which has concrete and abstract methods which will be declared using the abstract keyword, and it is implemented by other class using 'extends' keyword.

Can we create the object for abstract class?

No, we can't create the object for abstract class

What is the illegal combination of abstract class?

* Static abstract as static classes can't be overridden
* Final abstract as final class can't be changed
* Private abstract as this class needs to be implemented by other class and private restricts it

What is Interface in java?

* Interface is a special class which contains only abstract methods and public static final variables. From java 8 we can have static and default methods also in interface.
* For declaring interface, 'interface' keyword is used and it is implemented by other classes using 'implements' keyword.

What are the default members of interface?

* all the methods are by default public abstract methods
* all the variables are public static final

Can we create object or initiate the interface?

No, we can't instantiate the interface, that mean we can't create object, but reference can be called.

Can we write an interface without any member?

Through marker interface - example: random interface, serializable interface.

what is a marker or tagged interface?

* It is the interface that has no methods or constants inside it, it provides run-time type information about objects so compiler and JVM have additional information about the object.
* The same can be achieved using Annotations.

What is difference between interface and abstract class?

* If we don't know anything about implementation and we just have requirement specifications, we then go for interface and if we are talking about implementation but not completely then we go for abstract class.
* we can't declare the methods in interface with the modifiers like private, protected, final, static, there is no restriction for the modifiers in the methods of abstract class.
* we can't declare interface variable with the following modifiers: private, protected whereas there is no restriction in abstract class
* In interface we need to initialize the variable at the time of declaration itself, whereas in abstract class there is no need.
* In interface we can't declare instance and static block otherwise we get compile time error whereas in abstract class we can declare an instance and static blocks.
* In interface we can't declare a constructor whereas in abstract class we can declare and it will be executed at the time of child object creation

Can a class extend only one class?

yes

Can a class extend one or more interface?

No

Can an interface extend one or more interface?

Yes

Can an interface extend one or more class?

No

Can a class implement one or more class?

No

Can a class implement one or more interfaces?

Yes

Can an interface implement one or more class?

No

Can an interface implement one or more interface?

No

Can we define abstract class without abstract methods?

yes

Why we use interface?

To achieve abstraction fully because through abstract class we can't achieve full abstraction.

Can we write an inner class in an interface?

Yes

Can we declare local inner class as abstract?

Yes

Can we use abstract keyword with constructor, instance, initialization block, and static initialization block?

No

Why final and abstract can't be used together?

Both are opposite in nature. We use final when we don't want the method to be changed and abstract is used when we want the method to be implemented in child class.

Where do we use abstract class and interface in selenium?

* In selenium framework we use abstract in certain functionalities for better code structure and maintainability and reusability. example: creating findandclick function and thescreenshot function and have their concrete implementation in class which implements its parent class.
* Also public abstract class abstractwedrivereventlistener extends java.lang.object implments webdriverevenetlistner
* Interface example in selenium is webdriver

What are types of inheritance supported by java?

single, multilevel, hierarchical are supported by default and multiple and hybrid supported by interface

Why is multiple inheritance not supported in java?

* Multiple inheritance is concept of class inherit properties of more than one parent class and java doesn't support.
* Because when there are methods with same signature on super class and sub class, on calling the methods compiler can't determine which class method to be called and even on calling which class method gets the priority.
* In java multiple inherences is achieved by using default methods in interface.by this a class can implement 2 or more interfaces which are having default methods with same signature and the implementation class should explicitly specify which method is to be used or it should override the default method.

What is Method Overloading and Method overriding?

* method overloading means class having multiple methods with the same name, but the parameters count and type will vary
* method overriding means method in the parent class have same method in the child class and it is called at run time.

Can we overload by changing the return type?

no we can’t, we get compile time error

Can you override a private method in Java?

no private methods can't be overridden and inner class is allowed to access private data members of outer class

What is runtime polymorphism?

In the java at run time determines which method to be classes from the parent and child classes having same method

What is static binding and Dynamic binding?

when the type of object is determined at the compile type it is called static binding, if there is any final, static or private methods in a class there is static binding and when the object is determined at the run time it is called dynamic binding

What is Wrapper class in java?

* it provides the mechanism to convert primitive into object and object to primitive and conversion of primitive to object is call autoboxing and vice verse for unboxing
* As java is object-oriented programming and object is used everywhere so we need primitives to be converted to objects to use in java in cases like collections, serialization, synchronization and change the value in method
* primitive type Boolean, int, double, float, char, byte, short, long
* wrapper class: Boolean, character, byte, short, integer, long, float, double

what is difference between collection and collections?

* collection is an interface and collections is a class
* collection in java is the interface and root of java collection framework and most of the collections in java are inherited from this interface except map interface.
* collection - list, queue, set

What are the different types of collections available in Java?

List Interface in Java:

* contains ordered elements
* may include duplicates
* supports index-based search and random access but elements can be easily inserted irrespective of position.
* extended by linked list, array list

Queue in Java:

* follows a FIFO approach
* elements add at rear end and removed from front end
* Set in Java:
* doesn't define an order for elements
* doesn't support index-based search
* doesn't allow duplicate values
* hashset class, linked hashset class and extends by sorted set interface, tree set class.

Map in Java:

* Represents a key value pair
* map interface doesn't implement collection
* it can only contain a unique key it can have duplicate values

Array list in Java:

* it has dynamic resizing capability
* whenever the list is full, it can automatically resize to 50th of its original size.
* it is non synchronized.

LinkedList in Java:

* implements list and deque interface
* maintains the insertion order
* non synchronized
* doesn't support accessing elements randomly
* can use list iterator to iterate linked list elements

Vector in Java:

* synchronized and maintains insertion order
* thread safe and increases its size by doubling the array size
* it is legacy class

Hashset class:

* it implicitly implements a hashtable
* contains only unique elements
* only one null element can be added
* it is unordered as set

LinkedHashSet:

* ordered version of hashset which maintians a doubly linked list across all elements
* it preserves the insertion order

Sorted Set:

* All the elements are sorted set must implement the comparable interface.
* it is a set sorted in an ascending order

Tree set:

* uses a tree for storage
* objects in tree set are stored in a sorted and ascending order

what is difference between hashtable and hashmap and hashset?

Hashtable:

* it is synchronized in nature
* doesn't allow any null key or value

HashMap:

* It implements Map interface.
* it is non synchronized
* it allows only one null key but multiple null values
* It allows duplicate values but duplicate keys are not allowed
* It doesn't allow dummy values.
* It requires 1 objects during add operation.
* It uses Hashing technique for adding and storing mechanism
* It is comparatively faster than hashset because of hashing technique has been used here.
* It uses put() for insertion method.

HashSet:

* It implements set interface.
* It doesn't allow duplicates
* It allows dummy values
* It requires 2 objects during add operation.
* It uses Hashmap object technique for adding and storing mechanism
* It is comparatively slower than hashmap
* It have single null value
* It used add() for insertion method.

Which is preferred hash map or hash table?

Hash map is preferred most when there is no need of thread synchronization

Which map allows duplicate keys

MultiValuedmap

What is sorted Map in java?

* implicitly implements the red black tree implementation
* can't store any null key

Why doesn’t map extend the collection interface?

The map interface in java follows key value pair structures whereas collection interface is a collection of objects which are stored in a structured manner with a specified access mechanism. The main reason map doesn't extend the collection interface is that add(ele) method of the collection interface doesn't support the key value pair like map interface's put(k,v) method.

What are the types of iterators?

* For set and linked list for iterating through them we need to use iterators they are of 2 types fail fast and fail safe
* fail fast iterator throws concurrent modification exception when one thread is iterating iver collection object and other thread structurally modify collection either by adding or removing or modifying objects on underlaying collection.
* they are called fail fast because they try to immediately throw exception when they encountered failure
* fail safe iterator doesn't throw any exception if collection is modified structurally while one thread is iterating over it because they work on clone of collection instead of original collection and that’s why they are called as fail safe iterator

Why doesn’t hash table allow null and hashmap do?

* To successfully store and retrieve objects from hashtable the objects used as keys must implement the hash code method and equals methods.
* Since null is not an object it can't implement these methods.
* Hashmap is advanced version and improvement on hashtable.

Which one is faster Arraylist or Linkedlist?

ArrayList is fastest as it saves data according to indexes and it implements randomAccess interface provides the capability of random retrieval to arraylist whereas linkedlist doesn't implement randomAccess interface.

Which is preferred to use Arraylist or Linkedlist?

Arraylist is preferred over Linkedlist when there is no need to preserve order

How to sort an arraylist?

* collections.sort(list) - ascending order
* collections.sort(list,collections.reverseorder()); - descending order

What is synchronization and non-synchronization in java?

* synchronization basically means only one thread can access methods of that class at a time ex: string buffer
* synchronized collection method returns a thread safe collection backed up by specified collection
* synchronizationlist method is like above and used to create a synchronized list
* synchronizedmap , synchronizedset, synchronizedsortedset all these synchronized collections achieve thread safe through intrinsic locking and the entire collections are locked
* intrinsic locking is implemented via synchronized blocks within the wrapped collections methods.
* it assure data consistency / integrity in multi-threaded env but also have penalty of performance

What are Concurrent collections:

* They achieve thread safety by dividing their data into segments
* Concurrent collections are much more performant than synchronized collections due to inherent advantages of concurrent thread access.
* Synchronized collection means the class is thread safe and only one thread can access methods of that class at any given time ex: stringbuffer
* Nonsysnchronized collection mean than 2 or more threads can access the methods of that class at given time. it is thread safe

How to print the name of class?

system.out.println(this.getclass().getname())

What is difference between encapsulation and abstraction?

* Encapsulation means wrapping up a data under a single unit.
* it is the mechanism that binds together code and the data it manipulates.
* it is a protective shield that prevents the data from being accessed by code outside this shield.
* Technically in this, the variables or data of class is hidden from any other class and can be accessed only through any member function of own class in which they are declared.
* abstraction can also be called as data hiding.
* it can be achieved by declaring all the variables in the class as private and writing public methods in the class to set and get the values of variables.
* Data abstraction is the property by virtue of which only essential details are displayed to the user.
* the trivial or non-essentials units are not displayed to the user.
* it will be implemented using abstract class and interface.

What is difference between string and string buffer?

* string is immutable and string buffer is mutable
* string is slow and consumes more memory when you concat too many strings because every time it creates a new instance whereas string buffer is fast and consumes less memory when you concat strings
* string class overrides the equals method of the object class. so two strings can be compared using equals() method whereas stringbuffer doesn't override the equals() method of object class.

what is difference between Integer.valueOf() vs Integer.parseInt()

* integer.parseint() takes only string values where as other one takes both string and integer
* inter.parseint() returns primitive datatype int where as other one returns integer object

Which class is the superclass of all classes?

object is super class of any class by default

Why java is not 100% Object-oriented?

As it supports primitive datatypes like int, char etc. which are not objects.

What is Java Package and which package is imported by default?

package is collection of classes and interfaces. by default java.lang package is imported.

What are access modifiers?

* private: accessible only to that particular and can't be accessed out of the class
* default: accessible only under the package level
* protected: accessible within package and outside package to only child classes
* public: accessible everywhere

What is the base class for all the classes in java?

lang.object class is root or super class of the class hierarchy.

What is difference between == and equals()?

== checks if both objects point to the same memory location whereas equals evaluates to comparison of values in the objects

is null a keyword?

no it is literal and character string that is treated specially by the compiler if the compiler encounters it in a java source file.

Can we have try without catch block?

yes we can have and finally can be used where the block of code is executed even though the condition of try catch is not met

Can we have multiple public classes in a java source file?

no only one class among all the classes in the java file should be public. or else it throws compiler error

What is class level variable in java?

any variable declared with static modifier of which a single copy exists regardless of how many instances of class exist.

What is the default value for objects, string, Boolean, char, integers ,double in java?

object is null, string is null, byte is 0, short is 0, int is 0, long is 0L, float is 0.0f, double is 0.0d, char is \u0000 and boolean is false

What is final and finally, finalize difference?

final is keyword we use when we don't want the particular method or varaible not to be modfied. , finally is used along with try catch blocks to execute the code inspite of the condition, finalize is used to perform clean up processing just before object is garbage collected.

What does super keyword do?

it is reference variable which is used to refer immediate parent class object.

What is this keyword?

This keyword is used to refer to the current instance of the object

What is break and continue statement?

break is used when we want to exit from the condition and continue is used to break one iteration (in the loop) if a specified condition occurs and continues with the next iteration in the loop.

what is difference between throw and throws and throwable?

* throw keyword is used to throw an exception explicitly throws keyword is used to declare one or more exceptions, separated by commas. Throwable is super class of errors and exceptions in Java, only instances of this class or its sub class are thrown by java virtual machine or by the throw statement.
* If we want to create custom exceptions, then the class should extend the throwable class

What is typecasting?

process of assigning a primitive data type's value to another primitive data type

What is array?

it is collection of items stored at contiguous memory locations

How to sort array?

arrays.sort(array);

What is difference between Error and Exception?

both are subclasses of throwable class, errors mostly occur at run time that they belong to an unchecked type, exceptions are the problems which can occur at runtime and compile time.

What are the examples of java exceptions?

Arithmetic exceptions, arrayindexoutofbound exceptions, classnotfound exception, filenotfoundexception, nullpointerexception, numberformatexceptions, Stringindexoutofboundexceptions.

What are string, string buffer and string builder in Java?

* In java we have 3 classes to represent sequence of characters and string is immutable and string buffer and string builder are mutable.
* String buffer is synchronized (thread safe) so 2 threads can't call the methods of string buffer simultaneously whereas String builder is non-synchronized (not thread safe) and 2 threads can call the methods of string builder simultaneously.
* String builder is more efficient than string buffer.
* String can be used where the object is going to remain constant throughout the program and string builder can be used when we want string to access from single thread and string buffer when we want single to access from multiple threads.

How to convert string to string builder and string buffer and vice versa?

String to string builder and string buffer:

Text

Description automatically generated

What are oops concepts in Java?

* Object: An entity that has state and behavior is known as an object. An object can be defined as an instance of a class. It contains address and takes up some space inn memory.
* Class: It is Collection of objects. it can also be defined as blueprint from which we can create and individual object. it doesn’t consume any space.
* Inheritance: When one object acquires all the properties and behaviors of a parent object. It provides code reusability and is used to achieve runtime polymorphism.
* Polymorphism: It basically means existing in different forms and in java that can be achieved by method overloading and method overriding.
* Abstraction: It means hiding internal details and showing functionality
* Encapsulation: It is wrapping and binding up all the code together into a single unit.

What is the usage of a blank final variable?

* It is final variable that is not initialized during declaration.
* Blank final variables are used to create immutable objects (objects whose members can't be changed once initialized).
* Values must be assigned in constructor.
* If we have more than one constructor or overloaded constructor in class, then blank final variable must be initialized in all of them.
* constructor chaining can be used to initialize the blank final variable.

programs:

How to get the count of number of objects initialized in class?

constructor is used to create objects and when we count the number of times constructor is called we can get count of number of objects initialize

Text

Description automatically generated

program to rotate array:

Text

Description automatically generated

Text

Description automatically generated

Program to reverse array:

Text

Description automatically generated

Program to concatenate 3 variables to one variable in Java

Graphical user interface, text

Description automatically generated

Program to iterate list

Text

Description automatically generated

program to iterate the array

Text

Description automatically generated

program to iterate the map:

Text

Description automatically generated

Program to print how many times each character is repeated in string

Text

Description automatically generated

Program to find the duplicate words in string:

Text

Description automatically generated

program to find the factorial of number

Text

Description automatically generated

Program to reverse a string:

Text

Description automatically generated

Program to swap two variables without temp value:

Text

Description automatically generated

Text

Description automatically generated

Program to print sum of diagonals of matrix:

Text

Description automatically generated

program to check string is palindrome

Text

Description automatically generated

program to check number is palindrome

Text

Description automatically generated