What is the Java Virtual Machine and what is byte code?
Java Virtual Machine (JVM) executes Java byte code by translating it into machine code. Byte code (X.class) is a format produced by Java compiler after compiling the Java code (X.java).

How do you compile and run your java program without the help of an Integrated Development Environment (IDE) (e.g., an IDE like Eclipse)?
First, compile Java code into byte code using *javac X.java* and then execute that byte code on JVM using *java X*.

What is a main method in java? Explain signature parts of the main method?
The main method represents the starting point of a Java application/program. The signature of main method is *public static void main(String [] args)*. Here, public (access modifier) means this method is visible to other classes, static (keyword) means this method belongs to the class not object, void (return type) means this method returns nothing, main is the name of this method, and String [] args (parameter) means the array of arguments given through command line.

What are wrapper classes?
Classes that corresponds to primitive data types (int, long, double, boolean, etc.) are called wrapper classes. These classes are used to store primitive data types in collection classes such as List<Long> list.

What is a constructor?
A block of code that is used to create a new object of a particular class is called constructor. A class can have one or more constructors, and a constructor always comes after *new* keyword. For instance, Person p = new Person().

What is a package? Why is important to declare classes inside packages?
A package is a group of similar classes, interfaces, and sub-packages. Declaring classes inside packages is important because it allows the developer to group similar classes together. In addition, it also helps to modify the access level of classes, methods and fields.

Which access modifiers are available in Java? What is the default visibility or access level for classes, methods, and fields?
The available modifiers in Java are public, private, default (package-private), and protected. The default visibility for classes, methods and fields is at package level.

What is the difference between equals() and == ?
The equals() compares the states of the two variables/objects, whereas == operator compares the memory addresses of two variables/objects.

What is method overloading and method overriding?
Having two or more methods with same name and in same class but with different parameters is called method overloading. Having two methods with same name and parameters but with different implementation is called method overriding.

What are different ways to iterate over a list?
We can iterate over a list by using for loop, for each loop, while loop, iterator, and stream API.

What are the basic interfaces of Java Collections Framework?
Collection is a root interface that extends Iterable interface and provides core functionality required by any collection interface other than Map interface. List, Set, Queue, and Map are the basic interface of Java collection framework.

What it means by Generics in Java, why do we need Generics?
In Java, generics is a feature that enables the developers to write code (class/method) that works with different data types. We need generics because it allows us to implement generic collections as well as provide type-safety.

In the context of Java, what is an Exception? And what is an Error?
A disruption in the normal flow of execution due to some unexpected event that appear at runtime is called Exception. An internal (IOError) or external (hardware failure) exception that cannot be recovered is called Error.

What is unit testing? And what is JUnit?
Testing each component of a system in isolation is called unit testing. JUnit is a testing framework used by Java developers to perform unit testing.

How do I write a test that passes when an expected exception is thrown?
There are various ways, and one of the ways is to assert for expected exception. JUnit 5 provides assertThrows() for these cases. For instance,
assertThrows(NoSuchElementException.class, () -> this.sut.pop()).