**SQL**

1. **What is SQL**

Structured query language (SQL) is a programming language for storing and processing information in a relational database.  
I can use SQL statements to store, update, remove, search, and retrieve information from the database. You can also use SQL to maintain and optimize database performance.

(MySQL is an open-source relational database management system offered by Oracle.

MySQL is a popular database system for web applications. )

(refers to non-relational databases that don't use tables to store data)

1. **What is a foreign key**

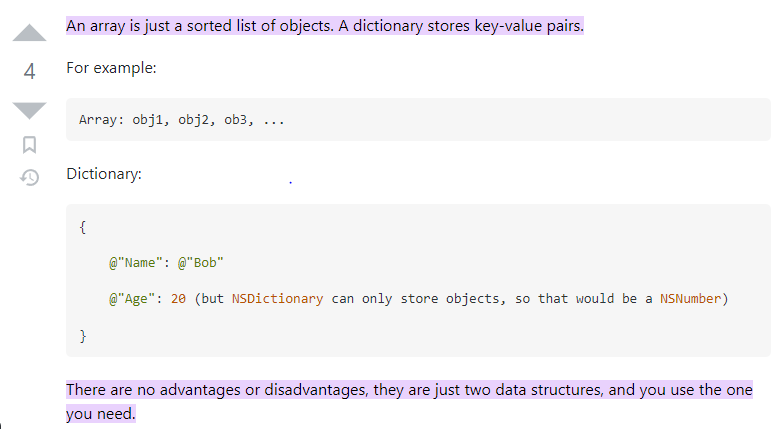
A foreign key (FK) is a column or combination of columns that is used to establish and enforce a link between the data in two tables to control the data that can be stored in the foreign key table.

1. **What is a Primary key?**

The primary key in SQL is a single, or a group of fields or columns that can uniquely identify a row in a table. Putting it simply, it is a column that accepts unique values for each row.

1. **What is the difference between an array and a dictionary**

Arrays are ordered collections of values. Sets are unordered collections of unique values. Dictionaries are unordered collections of key-value associations.



1. **What would you do if you needed a data element from a table in a SQL database?**

SELECT {attribute}+

FROM {table}+

[ WHERE {boolean predicate to pick rows} ]

[ ORDER BY {attribute}+ ];

1. **How do you join two tables together?**

A JOIN statement is used to combine rows from two or more tables, based on a related column between them.

Joining two tables in SQL can be done in four major ways: Inner Join (returns rows with matching columns), Left Join (ALL records in the left table and matching records in the right table), Right Join (ALL records in the right table and matching records in the left table), and Union (removes duplicates).

1. **What is a linked list?**Dynamic data structures which grow and shrink as data is added or deleted. They are linear and each element is a separate object.  
     
   \*An application that is dealing with an unknown number of objects will need to use a linked list.
2. **What is difference between a singly linked list and a doubly linked list?**Singly-Linked: Each node (or element) consists of 2 items - the data and a reference to the next node.  
     
   Doubly-Linked: Each node consists of 3 items - the data, a reference to the previous node and a reference to the next node.
3. **What are some advantages to linked lists**Easier to use when the amount of objects is unknown  
   • Insertion and deletion are much easier  
   • Only the exact amount of memory needed is used (so there is no waste)
4. **What are some disadvantages to linked lists**• If you need to access a specific node of the list, you have to start at the head and traverse through the whole thing.  
   • Reverse traversing is difficult unless you use a doubly linked list but then you'd need more memory for the pointer to previous  
   • Linked lists take up more memory than arrays because of the pointers to the next node
5. **What is an array?**A data type used to store multiple values in one variable. Arrays can be one-dimensional or multi-dimensional.
6. **What are some advantages to arrays**• An easy built-in way to store multiple values of the same type.  
   • Two-dimensional arrays can be used to store matrices.
7. **What are some disadvantages to arrays**• Must know in advance the amount of elements being stored.  
   • It's a static structure, once memory is allocated then it can't be increased or reduced.  
   • Each element is stored consecutively in memory so it's more difficult to insert and delete elements

**UNIX**

1. **What is unix**Unix is an operating system. It supports multi-tasking and multi-user functionality. Unix is most widely used in all forms of computing systems such as desktop, laptop, and servers. On Unix, there is a Graphical user interface similar to windows that support easy navigation and support environment.  
   It is a stable, multi-user, multi-tasking system for servers, desktops and laptops.
2. **PivotTables**  
   used for analysis and data reports, drag and drop into ROWS or COLUMNS, and then drag what values are under into VALUES

|  |  |
| --- | --- |
| ls | list files |
| ls -l | list files in long format |
| **ls –a** | list all files, including those beginning with a dot |
| **More (filename)** | shows first part of file |
| **Emacs (filename)** | editor that lets you create and edit a file |
| **Mv file file** | moves a file |
| **Cp file file** | copies a file |
| **Chmod (op file)** | lets you change the read, write, and execute permissions on file |
| **Mkdir (filename)** | make a new directory |
| **Cd (directory name)** | change directory |
| **pwd** | tells you where you currently are |
| **rlogin (hostname)** | lets you connect to a remote host |
| **ff** | find files anywhere on the system |

**OOP**

1. **What are the four pillars of object-oriented programming?**

**Object-oriented Programming (OOP)** is a programming paradigm based on the concept of "objects", which may contain data, in the form of fields, often known as attributes; and code, in the form of procedures, often known as methods.

**Abstration**Abstraction refers to showing only the essential features of the application and hiding the details. In C++/Java, classes provide methods to the outside world to access & use the data variables, but the variables are hidden from direct access. This can be done access specifiers. For example: phone call, we don't know the internal processing.  
(Say a man is driving a car. He only knows that the gas makes the car go and the brake stops it. He doesn't know (or need to know) how the process of accelerating or braking works just that it does.)

**Inheritance**Inheritance is a way to reuse code. The class which is inherited from, is called the base class, and the class which inherits the code from the base class is called a derived class. A derived class can use all the functions which are defined in the base class, making the code reusable.

**(**Say you have a "Car" class that defines methods like drive(), currentSpeed(), on/off, etc.  
You could then have a child class that extends Car, lets say "Ford" and it has total access to the parent functions and can also define it's own like turboBoost(), sportMode(), etc...)

**Encapsulation**Providing different levels of scope on how classes, methods, and attributes can be accessed and only allowing access on a need-to-know basis. Generally, oop uses private, public and protected keywords to implement encapsulation on attributes, methods and classes. For example, a private method in a class could only be accessed by the class and a public method could be accessed any other class..  
(Say there's a class "Shape." The variables length and width are declared as private so that if Shape is sub-classed and say "Rectangle" is the child, it won't be able to directly modify those variables. This makes reuse possible.)

**Polymorphism**It is a feature, which lets us create functions with same name but different arguments, which will perform differently. That is function with same name, functioning in different way. Or, it also allows us to redefine a function to provide its new definition.  
(Say there's a class "Animal" and a sub-class "Cat."  
The animal class would provide basic states and behaviors that all animals have and then each sub-class could define them on their own.  
EX: Cats and fish both move() but cats walk and fish swim.)

1. **Explain the algorithm of a pangram**
2. **What is do-while loop?**

The do/while loop is a variant of the while loop. This loop will execute the code block once, before checking if the condition is true, then it will repeat the loop as long as the condition is true.

1. **What is a tuple? What is TDD**

A table row contained in a table in the tablespace is known as a tuple. Typically, a table has rows and columns, where the rows represent records and the columns represent attributes. A tuple is a single row in a database that contains a single record for such a relation.

Test-driven development is a software development process relying on software requirements being converted to test cases before software is fully developed, and tracking all software development by repeatedly testing the software against all test cases.

1. **What is recursion? could you explain it?**

recursion is a method of solving a [computational problem](https://en.wikipedia.org/wiki/Computational_problem) where the solution depends on solutions to smaller instances of the same problem.[[1]](https://en.wikipedia.org/wiki/Recursion_(computer_science)#cite_note-1)[[2]](https://en.wikipedia.org/wiki/Recursion_(computer_science)#cite_note-2) Recursion solves such [recursive problems](https://en.wikipedia.org/wiki/Recursion) by using [functions](https://en.wikipedia.org/wiki/Function_(computer_science)) that call themselves from within their own code

(The process in which a function calls itself directly or indirectly)

1. **Difference between Object and class**Object is an instance of a class. Class is a blueprint or template from which objects are created. Object is a real world entity such as pen, laptop, mobile, bed, keyboard, mouse, chair etc. Class is a group of similar objects.
2. **What is an abstract class**A class that contains abstract methods (or methods without bodies.) They can only be sub-classed using the keyword "extends."  
   \*If the abstract class is sub-classed then the child only defines the methods it wishes to use.
3. **What is java**A programming language that's heavily based off of the syntax of C and C++ but has less low-level programming control. Java is a very general purpose OOP language and is compiled to byte-code and run on a JVM (Java Virtual Machine.) Because of this, Java is able to run across many different platforms making it easier to port to other machines.
4. **What is an object**Meant to represent real world objects and have two characteristics: a state and a behavior.  
     
   \*States = variables (name, color, breed)  
   \*Behaviors = methods (bark, eat, fetch)
5. **What is a class**The blueprint for an object. Classes define the various states and behaviors of an object.   
   \*Not all classes require a main() method.  
   \*Many objects can be created from one class.
6. **What is a constructor**A function in a class that can take different arguments and assign values to the various states of an object. Constructors are instance methods that typically have the same name as the class.  
   \***Constructors** are either set default or user-defined values.
7. **What is garbage collection**The process of freeing up memory that's no longer being used and destroying objects that are no longer reachable.
8. **Why garbage collection is important in java**It's important because without it, the programmer would be responsible for freeing up the memory and if they forget then the program might terminate due with an OutOfMemory error.  
   \*In C/C++ it's the job of the programmer to create and destroy objects. Many times the programmer neglects to destroy which may cause the memory to full up and terminate the program.
9. **What is an unreachable object**An object is unreachable if and only if it doesn't contain any reference to it.
10. **How would you make an object unreachable**1) Nullifying the reference variable  
    2) Re-assigning the reference variable  
    3) The object was created inside of a method  
    4) Island of Isolation  
    To Request Garbage Collection  
    • System.gc()  
    • Runtime.getRuntime().gc()
11. **SDLC**The SDLC develops and describes a detailed plan that includes stages, or phases, each with its own process and deliverables. It describes the entire development process, including all tasks involved in planning, developing, testing, and distributing a software product.
12. **Agile environment**Agile is a process in which a team can manage a project by dividing it into several phases. It involves regular interaction with stakeholders and continuous improvement and photography at all stages. The Agile approach starts with customers explaining how the final product will be used and what problem it will solve.
13. **Binary, quick sort algorithms**An algorithm is **a procedure used for solving a problem or performing a computation**.
14. **Do u know data structure and most familiar one**Data Structure can be defined as the group of data elements which provides an efficient way of storing and organising data in the computer so that it can be used efficiently. Some examples of Data Structures are arrays, Linked List, Stack, Queue, etc. Data Structures are widely used in almost every aspect of Computer Science i.e. Operating System, Compiler Design, Artifical intelligence, Graphics and many more.  
      
    **Overloading** occurs when two or more methods in one class have the same method name but different parameters.  
    **Overriding** occurs when two methods have the same method name and parameters. One of the methods is in the parent class, and the other is in the child class. Overriding allows a child class to provide the specific implementation of a method that is already present in its parent class.​