Assignment Day 1

Q1) Static Typed Language - In this type of language we can write and use variables without declaring them. Static variables variables have an immutable type, decided beforehand.

They can only be operated after conversion. If any language allows memory allocation at compilation time, then it is called Static typed Language -C, C++,

Dynamic Typed Language – In this type of language if we have to use a variable, we need to declare them. If any language allows memory allocation at Run time, then it is called Static typed Language- Python, java.

Q2) Programming Language - is a set of instructions that can be fed into a computer to achieve a specific output. They are compressed into smaller packages that do not need to be interpreted by another language or application. code when compiled creates binary code executable files also known as **.exe** files that take memory.

Scripting Language - is a programming language supporting scripts written exclusively for a special runtime environment to automate a specific action/function execution. Do not create .exe executable files.

* Q3) Imperative: Programming with an explicit sequence of commands that update state.
* Declarative: Programming by specifying the result you want, not how to get it.
* Structured: Programming with clean, goto-free, nested control structures.
* Procedural: Imperative programming with procedure calls.
* Functional (Applicative): Programming with function calls that avoid any global state.
* Function-Level (Combinator): Programming with no variables at all.
* Object-Oriented: Programming by defining objects that send messages to each other. Objects have their own internal (encapsulated) state and public interfaces. Object orientation can be:
  + Class-based: Objects get state and behaviour based on membership in a class.
  + Prototype-based: Objects get behaviour from a prototype object.
* Event-Driven: Programming with emitters and listeners of asynchronous actions.
* Flow-Driven: Programming processes communicating with each other over predefined channels.
* Logic (Rule-based): Programming by specifying a set of facts and rules. An engine infers the answers to questions.
* Constraint: Programming by specifying a set of constraints. An engine finds the values that meet the constraints.
* Aspect-Oriented: Programming cross-cutting concerns applied transparently.
* Reflective: Programming by manipulating the program elements themselves.
* Array: Programming with powerful array operators that usually make loops unnecessary.

Q4) HTTP – Hypertext transfer Protocol **is based on** the Client/Server model. Client/Server model can be explained as two computers, Client (receiver of service) and Server (provider of service) that are communicating via requests and responses.

HTTP2 is much faster and more reliable than HTTP1.1. HTTP1.1 loads a single request for every

TCP connection, while HTTP2 avoids network delay by using multiplexing.