**Assignemnet-1**

* **What are Red, White, Grey, Black Hat Hackers?**

**Red Hat**: A Red Hat hacker sometimes refers to a person who targets Linux based systems. However, in the hacking world, a Red Hat hacker plays a similar role to a White Hat hacker in protecting IT systems from cyberattacks but from a different perspective. They launch aggressive attacks against Black Hat to bring them down and destroy their computer and other resources.

**White Hat**: A white hat is an ethical computer hacker, or a computer security expert, who specializes in penetration testing and other testing methodologies that ensure the security of an organization's information systems. They only seek vulnerabilities/ exploits when they are legally permitted to do so.

**Black Hat**: Black Hat hackers are non-ethical hackers and criminals who break into computer networks with malicious intent. They may also release malware that destroys files, holds computers hostage, or steals passwords, credit card numbers, and other personal information.

**Gray Hat**: A grey hat is a computer hacker or computer security expert who may sometimes violate laws or typical ethical standards, but does not have the malicious intent typical of a black hat hacker. Gray hat hackers enact a blend of both black hat and white hat activities. Gray hat hackers often look for vulnerabilities in a system without the owner's permission or knowledge. If issues are found, they report them to the owner, sometimes requesting a small fee to fix the problem.

**2. Different Kinds of attack?**

1. Malware. Malware is a term used to describe malicious software, including spyware, ransomware, viruses, and worms.
2. Phishing.
3. Man-in-the-middle attack.
4. Denial-of-service attack.
5. SQL injection.
6. Zero-day exploit.
7. DNS Tunnelling.

**3. Interfaces in java?**

An interface in Java is a blueprint of a class. It has static constants and abstract methods. The interface in Java is a mechanism toachieve [abstraction](https://www.javatpoint.com/abstract-class-in-java). There can be only abstract methods in the Java interface, not method body. It is used to achieve abstraction and multiple [inheritance in Java](https://www.javatpoint.com/inheritance-in-java). In other words, you can say that interfaces can have abstract methods and variables. It cannot a method body. We should use an interface if you want a contract on some behaviour or functionality. We should not use an have interface if you need to write the same code for the interface methods. In this case, you should use an abstract class, define the method once, and reuse it as needed.

**4. What is lambda?**

Lambda expression is a new and important feature of Java which was included in Java SE 8. It provides a clear and concise way to represent one method interface using an expression. It is very useful in collection library. It helps to iterate, filter and extract data from collection. The Lambda expression is used to provide the implementation of an interface which has functional interface. It saves a lot of code. In case of lambda expression, we don't need to define the method again for providing the implementation. Here, we just write the implementation code. Java lambda expression is treated as a function, so compiler does not create .class file.

* To provide the implementation of Functional interface.
* Less coding.

Java lambda expression is consisted of three components.

**1) Argument-list:** It can be empty or non-empty as well.

**2) Arrow-token:** It is used to link arguments-list and body of expression.

**3) Body:** It contains expressions and statements for lambda expression.

The simplest lambda expression contains a single parameter and an expression

Parameter->expression, if more than one parameter, wrap them in parentheses: (parameter1, parameter2)->expression.

**5. What is #Pragma?**

The #pragma in C is a pre-processor and a directive that is provided by the C standard in order to provide extra required details to the C compiler. These extra details can be anything that was somehow not passed within the program or the code logic. These directives, known as pragma are prefixed by the STDC in the standard.

Syntax : #pragma token.