1. Ce este Internetul?
2. Cum funcționează comunicarea pe internet, de la un utilizator, printr-un client, spre un server și înapoi?
3. Care este diferența dintre http si https?
4. Ce este un API?
5. Ce este un algoritm?
6. Cum stim ca un algoritm este eficient?
7. Care este diferența dintre JRE, JVM și JDK?
8. Ce înseamnă compilarea codului Java?
9. Ce reprezintă un fișier cu extensia .jar?
10.  Care sunt cele două mari categorii de date care pot fi stocate în variabile?
11.  Ce constrângeri există pentru alegerea numelui unei variabile? Dar recomandări?
12.  De ce este nevoie pentru a putea folosi un obiect din memoria programului după crearea acestuia?
13. The internet is a global wide area network connecting millions of computers. The internet is decentralized and each internet computer, called a host, is independent. Internet is an acronym of INTERconected NETworks.

The internet uses TCP/IP to transmit data via various types of media. The internet is a network of global exchanges – including private, public, business, academic and government networks – connected by guided, wireless and fiber-optic technologies.

1. Because the internet is a global network of computers each computer connected to the internet must have a unique address. Internet addresses are in the form nnn.nnn.nnn.nnn where nnn must be a number from 0-255. This address is known as an IP address. When you are connected to the internet, your computer has a unique IP address. If you connect to the Internet through an Internet Service Provider (ISP), you are usually assigned a temporary IP address for the duration of your dial-in session. If you connect to the Internet from a local area network (LAN) your computer might have a permanent IP address or it might obtain a temporary one from a DHCP (Dynamic Host Configuration Protocol) server.
2. HTTP or HyperText Transfer Protocol is a protocol using witch hypertext is transferred over the Web. Due to its simplicity, HTTP has been the most widely used protocol for data transfer over the web but the data exchanged using http is not very secure. Anyone between the browser and server can read it relatively easy if one intercepts this exchange of data.

HTTPS was introduced so that a secure session is setup first between Server and Browser.

Another difference between http and https is that http uses default port 80 while https uses default port 443.

* In HTTP, URL begins with “http://” whereas URL starts with “https://”
* HTTP uses port number 80 for communication and HTTPS uses 443
* HTTP is considered to be unsecure and HTTPS is secure
* HTTP Works at Application Layer and HTTPS works at Transport Layer
* In HTTP, Encryption is absent and Encryption is present in HTTPS as discussed above
* HTTP does not require any certificates and HTTPS needs SSL Certificates

1. API or Application Programming Interface is a software intermediary that allows two applications to talk to each other. APIs are a set of functions and procedures that allow for the creation of applications that access data and features of other applications, services or operating system.

According to [Wikipedia](https://en.wikipedia.org/wiki/Application_programming_interface), good APIs makes it easier to develop a computer program by providing all the building blocks, which are then put together by the programmer.”

An application program interface (API) is a toolset that programmers can use to help them create software. To put it simply, an API specifies how software components should interact.

1. “An algorithm is a sequence of instructions telling a computer what to do” – Pedro Domingos.

Algorithm is a procedure that describes the exact steps needed for the computer to solve a problem.

1. There are several ways to say that an algorithm is efficient or not. The mostly used ways:

**Time efficiency -**a measure of amount of time for an algorithm to execute.

**Space efficiency**- a measure of the amount of memory needed for an algorithm to execute.

**Complexity theory -**a study of algorithm performance.

**Function dominance -**a comparison of cost functions

1. Care este diferența dintre JRE, JVM și JDK?

JVM (Java Virtual Machine) is an abstract machine. It is called a virtual machine because it doesn't physically exist. It is a specification that provides a runtime environment in which Java bytecode can be executed. It can also run those programs which are written in other languages and compiled to Java bytecode.

The JVM performs the following main tasks:

* Loads code
* Verifies code
* Executes code
* Provides runtime environment

JRE is an acronym for Java Runtime Environment. It is also written as Java RTE. The Java Runtime Environment is a set of software tools which are used for developing Java applications. It is used to provide the runtime environment. It is the implementation of JVM. It physically exists. It contains a set of libraries + other files that JVM uses at runtime.

JDK is an acronym for Java Development Kit. The Java Development Kit (JDK) is a software development environment which is used to develop Java applications and applets. It physically exists. It contains JRE + development tools.

The JDK contains a private Java Virtual Machine (JVM) and a few other resources such as an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (Javadoc), etc. to complete the development of a Java Application.

1. Ce înseamnă compilarea codului Java?

Compiling a Java program means taking the programmer-readable text in your program file (also called source code) and converting it to bytecodes, which are platform-independent instructions for the Java VM.

1. Ce reprezintă un fișier cu extensia .jar?

A JAR file is a Java archive (JAR) file used by the Java Runtime Environment (JRE), a framework used for executing [Java](https://techterms.com/definition/java) programs. JAR files may serve as program libraries or as standalone programs that run if the JRE is installed on the computer or mobile device.

1.  Care sunt cele două mari categorii de date care pot fi stocate în variabile?

There are two major categories of data types in the Java language: primitive types and reference types.

1. Rules:

* The first character must be a letter or an underscore (\_). You can't use a number as the first character.
* The rest of the variable name can include any letter, any number, or the underscore. You can't use any other characters, including spaces, symbols, and punctuation marks.
* As with the rest of JavaScript, variable names are case sensitive. That is, a variable named Interest\_Rate is treated as an entirely different variable than one named interest\_rate.
* There's no limit to the length of the variable name.
* You can't use one of JavaScript's reserved words as a variable name. All programming languages have a supply of words that are used internally by the language and that can't be used for variable names because doing so would cause confusion (or worse). Note, too, that JavaScript also has many keywords that should be avoided as well.

Ideas for good variable name:

* Make your name descriptive
* Because JavaScript is case sensitive, consider using all lowercase letters in your variable names. This ensures that you never run into errors because you misused uppercase and lowercase letters, plus it's easier on the typing fingers.
* Choose a name that describes WHAT the object does, instead of how it does it. Review your data structures and functions to avoid using word tricks to keep track of all your variables.

1. De ce este nevoie pentru a putea folosi un obiect din memoria programului după crearea acestuia?

JVM (Java Virtual Machine) asa cum ii si spune numele este o mașină abstractă. Se numește o mașină virtuală fiindca ea nu există fizic. Este o specificație care furnizează un mediu de rulare în care poate fi executat bytecode Java. Poate rula si acele programe care sunt scrise în alte limbi și compilate în bytecode Java. JVM afiseaza următoarele sarcini principale: Codul sarcinilor Verifică codul Execută codul Oferă mediu de rulare