

Web with Python questions

Software design

Clean code

Point out 5 suggestions, how to format an SQL query!

1. Use Uppercase for the Keywords: use uppercase for the SQL keywords, and lowercase for your tables and columns.
2. Use Snake Case for the schemas, tables, columns (snake_case)
3. Use aliases when it improves readability: aliases are a convenient way to rename tables or columns which doesn't make sense.
4. Formatting: Carefully use Indentation & White spaces: Ident after a keyword, and when you use a subquery or a derived table.
5. Avoid writing too many comments in the code.

What layers can you name in a simple web application?

- HTML is the markup language that we use to structure and give meaning to our web content, for example defining paragraphs, headings, and data tables, or embedding images and videos in the page.
- CSS is a language of style rules that we use to apply styling to our HTML content, for example setting background colors and fonts, and laying out our content in multiple columns.
- JavaScript is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else.

The three layers build on top of one another nicely.

Error handling

What error can occur, when an array does not have an element on the requested index?

IndexError

What is the "finally" block, and how would you use it?

- finally clause. The optional finally clause runs after the try block exits (whether or not this is due to an exception being raised). You can use it to clean up any resources, close files, etc. It is used for code that you want to run whether an exception occurred or not.

```
`    try:
        my_function(6, 2)
    except ZeroDivisionError as e:
        print(e)
    else:
        print('Everything worked OK')
    finally:
        print('Always runs')`
```

Why should we catch special exception types?

By catching an exception we can separate the details of what to do when something out of the ordinary happens from the main logic of a program.

Security

What is SQL injection? How to protect an application against it?

SQL injection is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.

Protection:

1. **Validate User Inputs:** A common first step to preventing SQL injection attacks is validating user inputs. First, identify the essential SQL statements and establish a whitelist for all valid SQL statements, leaving unvalidated statements out of the query. This process is known as input validation or query redesign (e-mail, phone number)
2. **Sanitize Data By Limiting Special Characters:** Doing this can ensure that any dangerous characters such as a single quote ' is not passed to a SQL query as instructions. A primary method of avoiding these unauthenticated queries is the use of prepared statements.
3. **Enforce Prepared Statements And Parameterization:** use prepared statements with parameterized queries, also known as variable binding, for writing all database queries. By defining all SQL code involved with queries, or parameterization, you can distinguish between user input and code.

What is XSS? How to protect an application against it?

Cross-site scripting (also known as XSS) is a web security vulnerability that allows an attacker to compromise the interactions that users have with a vulnerable application. XSS attacks enable attackers to inject client-side scripts into web pages viewed by other users. It allows an attacker to circumvent the same origin policy, which is designed to segregate different websites from each other. Cross-site scripting vulnerabilities normally allow an attacker to masquerade as a victim user, to carry out any actions that the user is able to perform, and to access any of the user's data. If the victim user has privileged access within the application, then the attacker might be able to gain full control over all of the application's functionality and data.

1. **Filter input on arrival.** At the point where user input is received, filter as strictly as possible based on what is expected or valid input.
2. **Encode data on output.** At the point where user-controllable data is output in HTTP responses, encode the output to prevent it from being interpreted as active content. Depending on the output context, this might require applying combinations of HTML, URL, JavaScript, and CSS encoding.
3. **Use appropriate response headers.** To prevent XSS in HTTP responses that aren't intended to contain any HTML or JavaScript, you can use the Content-Type and X-Content-Type-Options headers to ensure that browsers interpret the responses in the way you intend.

4. Content Security Policy. As a last line of defense, you can use Content Security Policy (CSP) to reduce the severity of any XSS vulnerabilities that still occur.

How to properly store passwords?

The best way to protect passwords is to employ salted password hashing.

What is HTTPS?

Hypertext transfer protocol secure (HTTPS) is the secure version of HTTP, which is the primary protocol used to send data between a web browser and a website. HTTPS is encrypted in order to increase security of data transfer. This is particularly important when users transmit sensitive data, such as by logging into a bank account, email service, or health insurance provider.

What is encryption and decryption?

Encryption is the process of translating plain text data (plaintext) into something that appears to be random and meaningless (ciphertext). Decryption is the process of converting ciphertext back to plaintext.

To encrypt more than a small amount of data, symmetric encryption is used. A symmetric key is used during both the encryption and decryption processes. To decrypt a particular piece of ciphertext, the key that was used to encrypt the data must be used.

What is hashing?

Hashing is the process of mapping any arbitrary size data into a fixed-length value using a hash function.

Hash algorithms are one way functions. They turn any amount of data into a fixed-length "fingerprint" that cannot be reversed. They also have the property that if the input changes by even a tiny bit, the resulting hash is completely different. This is great for protecting passwords, because we want to store passwords in a form that protects them even if the password file itself is compromised, but at the same time, we need to be able to verify that a user's password is correct.

```
`hash("hello") =  
2cf24dba5fb0a30e26e83b2ac5b9e29e1b161e5c1fa7425e73043362938b9824  
hash("hbllo") =  
58756879c05c68dfac9866712fad6a93f8146f337a69afe7dd238f3364946366`
```

What is the difference between encryption and hashing? When would you use which?

Hashing and encryption are the two most important and fundamental operations of a computer system. Both of these techniques change the raw data into a different format. Hashing on an input text provides a hash value, whereas encryption transforms the data into ciphertext.

Although both of these techniques convert the data into a separate format, there are subtle differences in their conversion mechanism and area of usage.

Encryption's purpose is to transmit or store data securely, whereas hashing is used to verify data. Hashing allows you to verify data is correct without needing to see it.

For example, when you enter a password into a website, the plain text will typically be converted to a fixed output using a hashing algorithm (also called a hash function). Instead of checking to verify the password itself is correct, the system checks to verify that the hashed output is correct. This way, the password is never actually revealed.

Unlike encryption, which is meant to be unscrambled using a key, a hash value is virtually impossible to revert back to the original input value.

What encryption methods do you know?

There are two fundamental types of encryption: symmetric encryption (which uses a single key) and asymmetric encryption (which requires two keys).

Symmetric encryption: A symmetric key encryption scheme uses a single symmetric key to both encrypt and decrypt data. The key needs to be shared with all authorized people. While symmetric encryption is much faster and less resource-intensive than its asymmetric counterpart, it's also less secure.

Symmetric encryption is useful when processing speed is important (payment transactions) or if you don't need to share the data with another party (personal backup drive).

Asymmetric encryption: Also called public key cryptography, this encryption method uses two separate keys. One key is made public (shared with everyone) and one key is kept private (known only to the key's generator). The public key is used to encrypt the data and the secret key is needed to decrypt it.

Asymmetric encryption offers more security than symmetric encryption, but it might be overkill for some purposes and its processes can slow down transactions, networks, and machines.

4 of the most common encryption methods

Different encryption methods are based on the type of keys used, key length, and size of data blocks encrypted. Here are some of the common encryption methods that you might see used in various encryption tools:

1. Advanced Encryption Standard (AES)

Advanced Encryption Standard is a symmetric encryption algorithm that encrypts fixed blocks of data (of 128 bits) at a time. The keys used to decipher the text can be 128-, 192-, or 256-bit long. The 256-bit key encrypts the data in 14 rounds, the 192-bit key in 12 rounds, and the 128-bit key in 10 rounds. Each round consists of several steps of substitution, transposition, mixing of plaintext, and more. AES encryption standards are the most commonly used encryption methods today, both for data at rest and data in transit.

2. Rivest-Shamir-Adleman (RSA)

Rivest-Shamir-Adleman is an asymmetric encryption algorithm that is based on the factorization of the product of two large prime numbers. Only someone with the knowledge of these numbers will be able to decode the message successfully. RSA is often used when transmitting data between two separate endpoints (e.g., web connections), but works slowly when large volumes of data need to be encrypted.

3. Triple DES (Data Encryption Standard)

Triple DES is a symmetric encryption and an advanced form of the DES method that encrypts blocks of data using a 56-bit key. Triple DES applies the DES cipher algorithm three times to each data block. Triple DES is commonly used to encrypt ATM PINs and UNIX passwords.

4. Twofish

Twofish is a license-free encryption method that ciphers data blocks of 128 bits. It's considered the successor to the 64-bit Blowfish encryption method and more versatile than its specialized successor, Threefish. Twofish always encrypts data in 16 rounds regardless of the key size. Though it works slower than AES, the Twofish encryption method continues to be used by some file and folder encryption software solutions.

What hashing methods do you know?

MD5 – An MD5 hash function encodes a string of information and encodes it into a 128-bit fingerprint. MD5 is often used as a checksum to verify data integrity. However, due to its age, MD5 is also known to suffer from extensive hash collision vulnerabilities, but it's still one of the most widely used algorithms in the world.
SHA-2 – SHA-2, developed by the National Security Agency (NSA), is a cryptographic hash function. SHA-2 includes significant changes from its predecessor, SHA-1. The SHA-2 family consists of six hash functions with digests (hash values) that are 224, 256, 384 or 512 bits: SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, SHA-512/256.
CRC32 – A cyclic redundancy check (CRC) is an error-detecting code often used for detection of accidental changes to data. Encoding the same data string using CRC32 will always result in the same hash output, thus CRC32 is sometimes used as a hash algorithm for file integrity checks. These days, CRC32 is rarely used outside of Zip files and FTP servers.

How/where would you store sensitive data (like db password, API key, ...) of your application?

Sensitive data should be always stored on a secure server that can only be accessed by people who are authorized to see it and never on the same server that hosts a website that is open to the public. The passwords used to access the data should be stored encrypted and the data itself should only be kept for as long as it's necessary.

Computer science

Algorithms

What is the difference between Stack and Queue data structure?
Stack and Queue both are the non-primitive data structures. The main differences between stack and queue are that stack uses LIFO (last in first out) method to access and add data elements whereas Queue uses FIFO (First in first out) method to access and add data elements.

Stack has only one end open for pushing and popping the data elements on the other hand Queue has both ends open for enqueueing and dequeuing the data elements.

What is BubbleSort? Describe the main logic of this sorting algorithm.

Bubble sort is a sorting algorithm that compares two adjacent elements and swaps them until they are not in the intended order. Just like the movement of air bubbles in the water that rise up to the surface, each element of the array moves to the end in each iteration. Therefore, it is called a bubble sort.

Suppose we are trying to sort the elements in ascending order.

First Iteration (Compare and Swap)

1. Starting from the first index, compare the first and the second elements.
2. If the first element is greater than the second element, they are swapped.
3. Now, compare the second and the third elements. Swap them if they are not in order.
4. The above process goes on until the last element.

Remaining Iteration

The same process goes on for the remaining iterations.

After each iteration, the largest element among the unsorted elements is placed at the end.

In each iteration, the comparison takes place up to the last unsorted element.

The array is sorted when all the unsorted elements are placed at their correct positions.

Explain the process of finding the maximum and minimum value in a list of numbers!

We initialize the minimum or maximum element to the first element and then traverse the array, comparing each element and update minimum and maximum whenever necessary.

Explain the process of calculating the average value in an array of numbers!

We need to find sum and divide sum by total number of elements.

What is Big O complexity? Explain time and space complexity!

Big O notation is a mathematical notation that describes the limiting behavior of a function when the argument tends towards a particular value or infinity. In computer science, big O notation is used to classify algorithms according to how their run time or space

requirements grow as the input size grows.

In plain words, Big O notation describes the complexity of your code using algebraic terms.

To understand what Big O notation is, we can take a look at a typical example, $O(n^2)$, which is usually pronounced "Big O squared". The letter "n" here represents the input size, and the function " $g(n) = n^2$ " inside the " $O()$ " gives us an idea of how complex the algorithm is with respect to the input size.

A typical algorithm that has the complexity of $O(n^2)$ would be the selection sort algorithm. Selection sort is a sorting algorithm that iterates through the list to ensure every element at index i is the i th smallest/largest element of the list.

Time complexity of the algorithms is that we only care about how much time it takes for the program to complete the task (how many times the statements are executed.). What also matters is the space the program takes to complete the task. The space complexity is related to how much memory the program will use, and therefore is also an important factor to analyze.

The space complexity works similarly to time complexity. For example, selection sort has a space complexity of $O(1)$, because it only stores one minimum value and its index for comparison, the maximum space used does not increase with the input size. Some algorithms, such as bucket sort, have a space complexity of $O(n)$, but are able to chop down the time complexity to $O(1)$.

Explain the process of calculating the average value in a list of numbers!

In a loop add each element's value to a single variable while counting the length of the array by incrementing an other variable at each cycle. After the loop divide the sum of the array elements' by the length of the array.

Procedural

How the CASE condition works in SQL?

The CASE statement goes through conditions and returns a value when the first condition is met (like an if-then-else statement). So, once a condition is true, it will stop reading and return the result. If no conditions are true, it returns the value in the ELSE clause.

If there is no ELSE part and no conditions are true, it returns NULL.

Syntax:

```
`CASE
    WHEN condition1 THEN result1
    WHEN condition2 THEN result2
    WHEN conditionN THEN resultN
    ELSE result
END;`
```

How the switch-case condition works in JavaScript?

The switch-case is basically a nicer looking version of an if-else statement chain, we give it a variable as a parameter and check it

against certain conditions. The code under the first "case" condition to match will be executed, or if there's no match, the default code will be executed. However the default code is optional. All "case" condition need to end with a break statement, except for the last one.

How to achieve a switch-case-like structure in Python?

In the current version of Python there is an equally functional match-case structure implemented, but beside that the simplest way is to use dictionaries.

Explain variable scoping in Python!

A variable is only available from inside the region it is created. This is called scope.

A variable created inside a function belongs to the local scope of that function, and can only be used inside that function.

Scope of variables: Not all variables can be accessed from anywhere in a program. The part of a program where a variable is accessible is called its scope. There are four major types of variable scope and is the basis for the LEGB rule. LEGB stands for Local -> Enclosing -> Global -> Built-in.

- Local: Whenever you define a variable within a function, its scope lies ONLY within the function. It is accessible from the point at which it is defined until the end of the function and exists for as long as the function is executing (Source). Which means its value cannot be changed or even accessed from outside the function.

- Enclosing: A special scope that only exists for nested functions. If the local scope is an inner or nested function, then the enclosing scope is the scope of the outer or enclosing function. This scope contains the names that you define in the enclosing function. The names in the enclosing scope are visible from the code of the inner and enclosing functions.

- Global: Whenever a variable is defined outside any function, it becomes a global variable, and its scope is anywhere within the program. Which means it can be used by any function.

- Built-in: All the special reserved keywords fall under this scope. We can call the keywords anywhere within our program without having to define them before use. Keywords are simply special reserved words. They are kept for specific purposes and cannot be used for any other purpose in the program. (e.g. if, else, elif, raise, pass, continue, False, None, in, from, for, True, while)

LEGB rule: The LEGB rule is a kind of name lookup procedure, which determines the order in which Python looks up names. For example, if you reference a given name, then Python will look that name up sequentially in the local, enclosing, global, and built-in scope. If the name exists, then you'll get the first occurrence of it.

Otherwise, you'll get an error.

What's the difference between const and var in JavaScript?

- var declarations are globally scoped or function scoped while (let and) const are block scoped.

- var variables can be updated and re-declared within its scope; (let variables can be updated but not re-declared;) const variables

can neither be updated nor re-declared.

- They are all hoisted to the top of their scope. But while var variables are initialized with undefined, (let and) const variables are not initialized.

- While var (and let) can be declared without being initialized, const must be initialized during declaration.

How the list comprehension looks like in Python?

List comprehension: It is an elegant way of defining and creating a list. List Comprehension allows us to create a list using for loop with lesser code. What normally takes 3-4 lines of code, can be compressed into just a single line.

```
list = [i for i in range(11) if i % 2 == 0]
```

How the "ternary expression" looks like in Python?

<OnTrue> if <Condition> else <OnFalse>

The three operands are written as x if c else y which reads as "return x if c else return y".

How the ternary expression looks like in JavaScript?

condition ? exprIfTrue : exprIfFalse

condition: An expression whose value is used as a condition.

exprIfTrue: An expression which is evaluated if the condition evaluates to a truthy value (one which equals or can be converted to true).

exprIfFalse: An expression which is executed if the condition is falsy (that is, has a value which can be converted to false).

```
```javascript
```

```
var age = 26;
```

```
var beverage = (age >= 21) ? "Beer" : "Juice";
```

```
console.log(beverage); // "Beer"
```

```
```
```

How to import a function from another module in Python?

The import statement(s) has to be on the top of the current module. We can import a whole module or a single object or function of module.

The module can be part of our repository, the python language or imported from external source.

Syntax:

```
```python
```

```
import function from module
```

```
```
```

How to import a function from another module in JavaScript?

In javascript if we want to import a function, the function needs to be exported in its source module first.

The import statement is similar to the python format, except the module name has to be in quotation marks.

Syntax:

```
```javascript
```

```
//source module
export function functionName (parameters) {}
//target module
import functionName from "module-name"
````
```

Functional

What is recursion?

The process in which a function calls itself directly or indirectly is called recursion and the corresponding function is called as recursive function.

Write a recursive function which calculates the Fibonacci numbers!

Recursive method:

```Python

```
def fibonacci_sequence(n):
 if n == 0 or n == 1:
 return n
 else:
 return fibonacci_sequence(n-1) + fibonacci_sequence(n-2)
...`
```

#### How to store a function in a variable in Python?

In Python, we can assign a function to a variable. And using that variable we can call the function as many as times we want. Simply assign a function to the desired variable but without () i.e. just with the name of the function. If the variable is assigned with function along with the brackets (), None will be returned.

```Python

```
def func():
```

```
    ...
```

```
var=func
```

```
var()
```

```
var()
```

```
````
```

#### List the ways of defining a callable logical unit in JavaScript!

```javascript

```
function myFunction(){
```

```
    ...
```

```
}
```

```
var myFunction = () => {
```

```
    ...
```

```
}
```

```
const myFunction = function(number ) { return number * number }
```

```
````
```

#### What is an event listener? How to attach one?

An event is an important part of JavaScript. A web page responds according to an event occurred. Some events are user generated and some are generated by API's. An event listener is a procedure in JavaScript that waits for an event to occur. The simple example of an event is a user clicking the mouse or pressing a key on the keyboard.

The `addEventListener()` is an inbuilt function in JavaScript which takes the event to listen for, and a second argument to be called whenever the described event gets fired. Any number of event handlers can be added to a single element without overwriting existing event handlers.

```
```element.addEventListener(event, listener, useCapture);```
```

How to trigger an event in JavaScript?

An event is an interaction between JavaScript and HTML. Events can be handled either through `addEventListener()` method or we can trigger events on individual components by defining specific JavaScript functions.

There are many types of event, but the common factor is that an event listener constantly and asynchronously waits for a condition to be met, like clicking on a button, pressing a key or loading in a page which causes the event to fire and the event handler function gets executed.

What is a callback function? Tell some examples of its usage.

A callback function is a function passed into another function as an argument, which is then invoked inside the outer function to complete some kind of routine or action.

Examples:

- event listeners
- Where callbacks really shine are in asynchronous functions, where one function has to wait for another function (like waiting for a file to load).

What is a Python decorator? How does it work? Tell some examples of its usage.

A decorator in Python is a function that takes another function as its argument, and returns yet another function. Decorators can be extremely useful as they allow the extension of an existing function, without any modification to the original function source code.

example of decorator:

```
```Python
def sampleDecorator(func):
 def addingFunction():
 # some new statements or flow control
 print("This is the added text to the
actual function.")
 # calling the function
 func()
```

```
return addingFunction
```

```
@sampleDecorator
def actualFunction():
 print("This is the actual function.")
```

```
...
 actualFunction()
```

Examples:

- Flask decorator
- SQL

#### What is the difference between synchronous and asynchronous execution?

Synchronous JavaScript: As the name suggests synchronous means to be in a sequence, i.e. every statement of the code gets executed one by one. So, basically a statement has to wait for the earlier statement to get executed.

Asynchronous JavaScript: Asynchronous code allows the program to be executed immediately where the synchronous code will block further execution of the remaining code until it finishes the current one. This may not look like a big problem but when you see it in a bigger picture you realize that it may lead to delaying the User Interface.

## Programming languages

### SQL

#### How can you connect your application to a database server? What are the possible ways?

It can be achieved by using a database management system, like PostgreSQL or through an API. In PostgreSQL you have to create and manage your own database, but with API you can access any database which is connected to the web.

#### When do you use the DISTINCT keyword in SQL?

The SELECT DISTINCT statement is used to return only distinct (different) values.

Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values.

#### Talk about the behavior/goal of these base SQL clauses: WHERE, GROUP BY, HAVING, ORDER BY?

1. The WHERE clause is used to filter records. It is used to extract only those records that fulfill a specified condition.

2. The GROUP BY statement groups rows that have the same values into summary rows, like "find the number of customers in each country".

The GROUP BY statement is often used with aggregate functions (COUNT(), MAX(), MIN(), SUM(), AVG()) to group the result-set by one or more columns.

3. The HAVING clause was added to SQL because the WHERE keyword cannot be used with aggregate functions.

4. The ORDER BY keyword is used to sort the result-set in ascending or descending order. The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

#### What are aggregate functions in SQL? Give 3 examples.

- COUNT counts how many rows are in a particular column.
- SUM adds together all the values in a particular column.
- MIN and MAX return the lowest and highest values in a particular column, respectively.
- AVG calculates the average of a group of selected values.
- ARRAY\_AGG function is an aggregate function that accepts a set of values and returns an array in which each value in the set is assigned to an element of the array.
- STRING\_AGG

#### What kind of JOIN types do you know in SQL? Could you give examples?

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

- (INNER) JOIN: Returns records that have matching values in both tables
- LEFT (OUTER) JOIN: Returns all records from the left table, and the matched records from the right table
- RIGHT (OUTER) JOIN: Returns all records from the right table, and the matched records from the left table
- FULL (OUTER) JOIN: Returns all records when there is a match in either left or right table

#### What are the constraints in sql?

SQL constraints are used to specify rules for the data in a table. Constraints are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the table. If there is any violation between the constraint and the data action, the action is aborted.

The following constraints are commonly used in SQL:

- NOT NULL - Ensures that a column cannot have a NULL value
- UNIQUE - Ensures that all values in a column are different
- PRIMARY KEY - A combination of a NOT NULL and UNIQUE. Uniquely identifies each row in a table
- FOREIGN KEY - Prevents actions that would destroy links between tables
- CHECK - Ensures that the values in a column satisfies a specific condition
- DEFAULT - Sets a default value for a column if no value is specified
- CREATE INDEX - Used to create and retrieve data from the database very quickly

#### What is a cursor in SQL? Why would you use one?

The database cursor is a mechanism that enables traversal over the

records in a database. The cursor is used to retrieve, add or remove data from the database in a sequential manner.

#### What are database indexes? When to use?

A database index is a pointer to data in a table. Indexes are a powerful tool used in the background of a database to speed up querying. They power queries by providing a method to quickly lookup the requested data.

#### What are database transactions? When to use?

A database transaction is a unit of work performed within a database management system and treated in a coherent and reliable way independent of other transactions. A transaction generally represents any change in a database.

#### What kind of database relations do you know? How to define them?

- One-to-one: Both tables can have only one record on each side of the relationship. Each primary key value relates to no more than one record in the related table.
- One-to-many: The primary key table contains only one record that relates to none, one, or many records in the related table.
- Many-to-many: Each record in both tables can relate to none or any number of records in the other table. These relationships require a third table, called an associate or linking table.
- 

#### You have a table with an "address" field which contains data like "3525, Miskolc, Régiposta 9." (postcode, city, street name and address). How would you query all records related to Miskolc?

```
```SQL
SELECT * FROM table
HAVING CHARINDEX('Miskolc', address) <> 0
```
```

#### How would you keep track of what kind of data has changed after an UPDATE or DELETE operation in a table?

At the basic database level you can track changes by having a separate table that gets an entry added to it via triggers on UPDATE/DELETE statements.

### HTML & CSS

#### What's the difference between XML, XHTML and HTML?

XML is a markup language where all documents must be marked up correctly.

XHTML is a stricter, more XML-based version of HTML.

XHTML was developed to make HTML more extensible and flexible to work with other data formats (such as XML). In addition, browsers ignore errors in HTML pages, and try to display the website even if it has some errors in the markup. So XHTML comes with a much stricter error handling.

#### How to include a JavaScript file in a webpage?

```
`<script type="text/javascript" src="path-to-javascript-
```

```
file.js"></script>`
```

#### How to include a CSS file in a webpage?

```
`<head>
 <link rel="stylesheet" href="styles.css">
</head>`
```

#### How to select an element using its id in CSS?

```
`#id`
```

#### How to select elements using their class in CSS?

```
` .class`
```

#### How to select elements which have the 'alpha' and 'beta' classes in CSS?

```
` .alpha .beta`
```

#### How to select all list items in all ordered lists on the page in CSS?

```
`ol li`
```

#### How to select elements using their attributes in CSS?

```
`element[attribute="value"]`
```

#### What are UX and UI?

UX: The user experience (UX or UE) is how a user interacts with and experiences a product, system or service. It includes a person's perceptions of utility, ease of use, and efficiency. User experience (UX) focuses on having a deep understanding of users, what they need, what they value, their abilities, and also their limitations. It also takes into account the business goals and objectives of the group managing the project.

UI: The user interface (UI) is the point of human-computer interaction and communication in a device. This can include display screens, keyboards, a mouse and the appearance of a desktop. It is also the way through which a user interacts with an application or a website.

![(/Users/bogirencsenyi/projects/ui-vs-ux.png)]

#### Please list some points that an application should fulfill to have good UX.

1. Simplicity
2. Usability
3. Visual aesthetics

1. Visibility of system status. Users should always be informed of what the system, or product is doing, in a reasonable timeline.

2. Error prevention. Providing feedback when an error occurs is important, but better yet if you can help prevent an error from occurring in the first place. Providing smart defaults, and confirmation is beneficial.

3. Aesthetic and minimalist design. Don't overwhelm users by displaying unnecessary information. Keep screens and dialogues

focused and minimal to maximize visibility and clarity.

#### #### What is XML, XSLT, DTD?

Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Extensible Stylesheet Language Transformations (XSLT) is an XML-based language used, in conjunction with specialized processing software, for the transformation of XML documents.

A document type definition (DTD) is a set of markup declarations that define a document type for an SGML-family markup language (GML, SGML, XML, HTML). A DTD defines the valid building blocks of an XML document.

#### #### What is the difference between HTML and XML?



### ### Javascript

#### #### What is javascript?

JavaScript is a dynamic programming language that's used for web development, in web applications, for game development, and lots more. It is the third layer of the layer cake of standard web technologies. It allows you to implement dynamic features on web pages that cannot be done with only HTML and CSS.

#### #### When to use AJAX? Bring examples of its usage.

AJAX stands for Asynchronous JavaScript And XML. AJAX's most appealing characteristic is its "asynchronous" nature, which means it can communicate with the server, exchange data, and update the page without having to refresh the page.

The two major features of AJAX allow you to do the following:

- Make requests to the server without reloading the page
- Receive and work with data from the server

Examples:

- Comments
- Filtering data
- Form validation
- Text hints and autocomplete text boxes

#### #### What is DOM and how to manipulate it from Javascript?

The Document Object Model (DOM) is an application programming interface (API) for manipulating HTML and XML documents.

The DOM represents a document as a tree of nodes. It provides API that allows you to add, remove, and modify parts of the document effectively.

Note that the DOM is cross-platform and language-independent ways of manipulating HTML and XML documents.

Manipulate form JS:

-Selecting elements:

- getElementById() – select an element by id.
- getElementsByName() – select elements by name.



getElementsByTagName() – select elements by a tag name.  
getElementsByClassName() – select elements by one or more class names.  
querySelector() – select elements by CSS selectors.  
–Manipulating elements:  
createElement() – create a new element.  
appendChild() – append a node to a list of child nodes of a specified parent node.  
textContent – get and set the text content of a node.  
innerHTML – get and set the HTML content of an element.  
append() – insert a node after the last child node of a parent node.  
replaceChild() – replace a child element by a new element.  
removeChild() – remove child elements of a node.  
–Manipulating Element's Styles:  
style property – get or set inline styles of an element.  
getComputedStyle() – return the computed style of an element.  
className property – return a list of space-separated CSS classes.  
classList property – manipulate CSS classes of an element.  
Element's width & height – get the width and height of an element.

#### What are events and how/why to use them in Javascript?

Javascript has events to provide a dynamic interface to a webpage. An event is an action that occurs in the web browser, which the web browser feeds back to you so that you can respond to it. For example, when users click a button on a webpage, you may want to respond to this click event by displaying a dialog box. Each event may have an event handler which is a block of code that will execute when the event occurs. An event handler is also known as an event listener. It listens to the event and executes when the event occurs. Objects that can fire events have an addEventListener() method, that takes at least two arguments: the name of the event and a function to handle the event.

#### What is event bubbling/capturing? How would you use it?

Event flow explains the order in which events are received on the page from the element where the event occurs and propagated through the DOM tree.

There are two main event models: event bubbling and event capturing.

–In the event bubbling model, an event starts at the most specific element and then flows upward toward the least specific element (the document or even window).

–In the event capturing model, an event starts at the least specific element and flows downward toward the most specific element.

Event bubbling is the order in which event handlers are called when one element is nested inside a second element, and both elements have registered a listener for the same event (a click, for example). With bubbling, the event is first captured and handled by the innermost element and then propagated to outer elements.

With capturing, the event is first captured by the outermost element

and propagated to the inner elements.

How to use:

```
<element>.addEventListener(<eventName>, <callbackFunction>,
{capture : boolean});
```

#### What is JSON and how do we use it?

JSON – JavaScript Object Notation – is a set of text formatting rules for storing and transferring data in a machine and human readable way. It looks a lot like the object literal syntax of JavaScript, and it is from there JSON originates.

But JSON is not JavaScript. Officially it's a totally different language with its own spec but it plays such a big part in JavaScript development that it's important to cover.

JSON is used to transfer information – between your browser to a server, or saved in text files for retrieval later – because it's simply text. That means you can't store complex data like a function, but you can store arrays, objects containing simple data, strings and numbers.

Some JSON:

```
{ "name": "Yoda", age: 894, "lightsaber" : { "color": "green" }
Using JSON
```

Data is either converted to or from JSON, using methods called stringify and parse respectively. JSON is an object available in pretty much all modern browsers but there are ways of adding to a browser that doesn't have it.

```
`
 var jsonString = JSON.stringify({
 make: "McLaren",
 model: "MP4-12C",
 miles: 5023
 });`
```

JSON.stringify converts an object into a JSON string. In this example, jsonString becomes {"make": "McLaren", "model": "MP4-12C", "miles": 5023 }.

```
`
 var car = JSON.parse(jsonString);`
```

The string can then be converted back to a JavaScript object using JSON.parse. car is now usable as a normal JavaScript object, so you can set its properties:

```
`
 car.model = "P1";`
```

## Software engineering

### Version control

#### What type of branching strategy would you use?

Feature Branching Strategy is the one I understand the most, in which each specific feature/task/problem/user story gets a separate branch. This allows the team members to work separately in parallel. The feature branches must not live longer than necessary, that is the completion of the feature. It's also important to merge early and often.

#### What would you do if you find a bug on the production code (master branch)?

First, I try to understand the nature of the bug: when/how/why does it trigger, how should it work properly and who is responsible for it, just because he/she may know more about it. Then if I don't have any idea how to fix it, I try to find someone in the team who can and entrust fixing it to him/her. Later I may make sure if it is fixed.

#### How can you move changes from one branch to another in GIT?

git checkout [another branch]

git merge [a branch]

#### How does a VCS help with code reviews?

- Anyone who can access the repository can check the code on his/her own device
- VCS makes it possible to compare different version of the code from different branches commit to commit
- Everyone can see clearly who wrote what

#### What is your favorite git command? Why?

Push, is means success.

#### What does remote/local mean in Git?

Local repositories reside on the computers of team members. In contrast, remote repositories are hosted on a server that is accessible for all team members – most likely on the internet or on a local network.

Technically, a remote repository doesn't differ from a local one: it contains branches, commits, and tags just like a local repository. However, a local repository has a working copy associated with it: a directory where some version of your project's files is checked out for you to work with.

A remote repository doesn't have such a working directory: it only consists of the bare ".git" repository folder.

### DevOps

#### Why is it good to use a package manager software?

A package manager software makes installing, updating, configuring or removing applications simpler.

#### Why is it good to use a virtual environment for a project?

A virtual environment is a tool that helps to keep dependencies required by different projects separate by creating isolated python virtual environments for them. Virtual environments are isolated Python environments that contain all the packages and dependencies required to run a Python project. Virtual environments keep dependencies separated by project and because of that, they allow you to use different versions of Python or third party libraries in each project you work on.

### ### Networks

#### What kind of HTTP status codes do you know?

1xx informational response – the request was received, continuing process

2xx successful – the request was successfully received, understood, and accepted

3xx redirection – further action needs to be taken in order to complete the request

4xx client error – the request contains bad syntax or cannot be fulfilled

5xx server error – the server failed to fulfil an apparently valid request

#### What is a API?

API is the acronym for Application Programming Interface, which is a software intermediary that allows two applications to talk to each other by HTTP requests.

#### What is REST API?

A REST API is an application programming interface that conforms to the constraints of representational state transfer (REST) architectural style and allows for interaction with RESTful web services.

#### What is JSON? When to use?

JavaScript Object Notation (JSON) is a standard text-based format for representing structured data based on JavaScript object syntax. It is commonly used for transmitting data in web applications.

#### What is TCP/IP? What layers does it define, what are they responsible for?

TCP/IP, in full Transmission Control Protocol/Internet Protocol, standard Internet communications protocols that allow digital computers to communicate over long distances. The TCP/IP model consists of 4 layers:

- Network Access Layer: defines the mechanism of how data is physically transmitted via the network
- Internet Layer: send data packets to their destination networks
- Transport Layer: offer safe and reliable data transmission from a source system process to a destination system process
- Application Layer: defines the procedure of interaction between the host applications and the protocols

#### What's the difference between TCP and UDP?

- TCP stands for Transmission Control Protocol and UDP stands for User Datagram Protocol
- TCP is connection-oriented, while UDP is a connectionless internet protocol
- TCP is comparatively slower than UDP
- TCP has built-in systems to check for errors and to guarantee data will be delivered in the order it was sent, while with UDP error-checking and recovery services are not required
- TCP is perfect for transferring information like still images,

data files, and web pages, while UDP is ideal for real-time communications like broadcast or multitask network transmission

#### How does an HTTP Request look like? What are the most relevant HTTP header fields?

A HTTP Request consists of a method, host, path, HTTP version and 3 optional parts the headers, query string and request body. Some common HTTP request header fields are User-Agent, Accept, Content-Type and Cookie.

#### How does an HTTP Response look like? What are the most relevant HTTP header fields?

A HTTP Response consists of a HTTP version and a status code on the first line, then the HTTP headers on the subsequent lines and the final block is a data block, which contains the optional data. Some relevant HTTP response header fields are Server, Connection, Date, Age and Set-Cookie.

#### What is DNS? How does it work?

Domain Name System (DNS) is the phonebook of the Internet, it translates domain names to IP addresses so browsers can load Internet resources.

#### What is a web server?

The term web server can refer to hardware or software, or both of them working together.

On the hardware side, a web server is a computer that stores web server software and a website's component files. (for example, HTML documents, images, CSS stylesheets, and JavaScript files) A web server connects to the Internet and supports physical data interchange with other devices connected to the web.

On the software side, a web server includes several parts that control how web users access hosted files. At a minimum, this is an HTTP server. An HTTP server is software that understands URLs (web addresses) and HTTP (the protocol your browser uses to view webpages). An HTTP server can be accessed through the domain names of the websites it stores, and it delivers the content of these hosted websites to the end user's device.

At the most basic level, whenever a browser needs a file that is hosted on a web server, the browser requests the file via HTTP. When the request reaches the correct (hardware) web server, the (software) HTTP server accepts the request, finds the requested document, and sends it back to the browser, also through HTTP. (If the server doesn't find the requested document, it returns a 404 response instead.)

Basic representation of a client/server connection through HTTP

To publish a website, you need either a static or a dynamic web server.

A static web server, or stack, consists of a computer (hardware) with an HTTP server (software). We call it "static" because the server sends its hosted files as-is to your browser.

A dynamic web server consists of a static web server plus extra software, most commonly an application server and a database. We call it "dynamic" because the application server updates the hosted files before sending content to your browser via the HTTP server.

For example, to produce the final webpages you see in the browser, the application server might fill an HTML template with content from a database. Sites like MDN or Wikipedia have thousands of webpages. Typically, these kinds of sites are composed of only a few HTML templates and a giant database, rather than thousands of static HTML documents. This setup makes it easier to maintain and deliver the content

#### Explain the client-server architecture.

It is an architecture of a computer network in which many clients request and receive service from a centralized server. Client computers provide an interface to allow a computer user to request services of the server and to display the results the server returns. Servers wait for requests to arrive from clients and then respond to them.

#### What would you use a session for?

Use a session when you don't want the public to have access to your user variables. For example authentication data.

#### What would you use a cookie for?

Use cookies when the user information to be stored is relatively small and it's no problem if the user can see or delete it. For example user preferences.

## ## Software Development Methodologies

#### What kind of software development methodologies do you know? What are the main features of these?

1. Waterfall
2. Agile
3. Scrum

#### What are the SCRUM roles?

- product owner
- developers
- scrum master

#### What are the SCRUM ceremonies?

Sprint Planning Meeting

The goal of Sprint Planning is to answer the questions "What are we going to work on, and how are we going to do it?" It's also important for the team to have a shared goal and a shared commitment to this goal before beginning their Sprint – the list of items the team plans to work on during that specific Sprint. The team then breaks down these items into tasks, typically no bigger than a 2 days' worth of work.

Daily Stand-up Meeting

Once we begin a Sprint, we have what we call a Daily Scrum every day. Organized by the Scrum Master, Daily Scrum is typically a 15-minute stand-up meeting to synchronize the work of team members, i.e. what's done on the prior day, what needs to be done today, identify any impediments, and creates visibility around the work that everyone is doing in the Sprint.

#### Sprint Review Meeting

Held at the end of each sprint to demonstrate the added functionality. The goal is to get feedback from the product owner and other stakeholders to ensure that the delivered increment met the business need and to revise the Product Backlog based on the feedback. This feedback will then become items that will be looped back into the Product Backlog, where it can be ordered and pulled in by the team in a future Sprint.

#### Sprint Retrospective Meeting

Retrospectives typically last 90 minutes and are there to help us incorporate continuous improvement into our team culture and into our Sprint cadence. This is where the Scrum Team meets to reflect on their previous Sprint and to figure out how to improve as a team by asking – what went well, what did not and what can be improved. It allows the team to focus on its overall performance and identify strategies for continuous improvement.

#### #### What are the SCRUM artifacts?

Scrum uses three artifacts to help manage work. All three are defined and described below.

#### Product Backlog

The product backlog is an ordered list of everything that is known to be needed in a product based on the product goal. It is constantly evolving and is never complete.

#### Sprint Backlog

The sprint backlog is a list of everything that the team commits to achieve in a given sprint. Once created, no one can add to the sprint backlog except the development team.

#### Potentially Releasable Product Increment

At the end of every sprint, the team delivers a product increment that is potentially releasable, meaning that it meets their agreed-upon definition of done. (An example might be fully tested and fully approved.)

#### #### What is the main goal of a retrospective meeting?

The team members can reflect on their performance in hindsight of the finished project, thus gaining valuable knowledge that can be useful in future projects.

Sprint retrospectives focus on the process. During a sprint retrospective, the scrum team discusses what went right and areas for improvement in the sprint. They make tangible plans for how to improve their own process, tools and relationships.

#### Explain, when would you recommend to use the waterfall methodology?

Waterfall methodology is linear and doesn't allow flexibility. It prevails when the project has clear objectives, strict requirements, fixed deadline and budget.