# Az objektum-orientált JavaScript alapjai

Ez az oktatóanyag célja, hogy segítse azokat, akik alapszintű JavaScript-tudással rendelkeznek, hogy ráugorjanak az objektum-orientált programozásra.

## Feladatok

### Készítsen programozási nyelvet

Határozzon meg egy constructProgrammingLanguage() nevű függvényt.

1. A function.js fájlban található egy constructProgrammingLanguage() függvény.
2. A függvény egy új objektumot ad vissza a következő kulcsokkal: name, isStaticallyTyped és supportsOOP.
3. A függvény paramétereket kap a felépített objektum minden kulcsához (tehát name, isStaticallyTyped és supportsOOP).
4. Ha valamelyik billentyű nincs megadva függvényparaméterként, vagy nem a kívánt típus, a függvény új hibát dob ( név egy karakterlánc, az isStaticallyTyped egy logikai érték, a supportsOOP pedig egy logikai érték).

### Készítsen kódszerkesztőt

Határozzon meg egy constructCodeEditor() nevű függvényt.

1. A function.js fájlban található egy constructCodeEditor() nevű függvény.
2. A függvény egy új objektumot ad vissza a következő kulcsokkal: name és supportedLanguages.
3. A supportLanguages kulcs alapértelmezett értéke egy üres tömb.
4. A függvény kap egy paramétert a felépített objektum nevével.
5. Ha a név nincs megadva függvényparaméterként, vagy nem karakterlánc, a függvény új hibát dob.

### Adjon hozzá egy programozási nyelvet a szerkesztőhöz

Határozzon meg egy addProgrammingLanguageToEditor() nevű függvényt.

1. A function.js fájlban található egy addProgrammingLanguageToEditor() függvény.
2. A függvény két paramétert kap, az egyik egy programnyelvi objektum, a másik pedig egy kódszerkesztő.
3. Mindkét paraméternek objektumnak kell lennie (ellenkező esetben hibát dob).
4. A kódszerkesztő objektumnak rendelkeznie kell egy supportLanguages nevű kulccsal.
5. A függvény hozzáadja a kapott programozási nyelvet a kódszerkesztő objektum supportLanguages kulcsához/tömbjéhez.

### Távolítson el egy programozási nyelvet a szerkesztőből

Határozza meg a removeProgrammingLanguageFromEditor() nevű függvényt.

1. A function.js fájlban található egy removeProgrammingLanguageFromEditor() függvény.
2. A függvény két paramétert kap, az egyik egy programnyelvi objektum, a másik pedig egy kódszerkesztő.
3. Mindkét paraméternek objektumnak kell lennie (ellenkező esetben hibát dob).
4. A kódszerkesztő objektumnak rendelkeznie kell egy supportLanguages nevű kulccsal.
5. A funkció eltávolítja a kapott programozási nyelvet a kódszerkesztő objektum supportLanguages kulcsából/tömbjéből.

### Programozási nyelv osztály

Határozzon meg egy ProgramingLanguage nevű osztályt.

1. Az oop.js fájlban található egy ProgramingLanguage nevű osztály.
2. Az osztálynak van neve, isStaticallyTyped és supportsOOP tulajdonsága.
3. Az osztálynak van egy konstruktor metódusa, amely minden tulajdonsághoz paramétereket kap (tehát name, isStaticallyTyped és supportsOOP).
4. Ha valamelyik tulajdonság nincs megadva konstruktor paraméterként, vagy nem a kívánt típus, akkor a metódus új hibát dob (a név karakterlánc, az isStaticallyTyped logikai érték, a supportsOOP pedig logikai érték).

### Kódszerkesztő osztály

Határozzon meg egy CodeEditor nevű osztályt.

1. Az oop.js fájlban van egy CodeEditor nevű osztály.
2. Az osztálynak két tulajdonsága van: name és supportedLanguages.
3. Az osztálynak van egy konstruktor metódusa, amely paramétert kap a név tulajdonsághoz.
4. Ha a név nincs megadva konstruktor paraméterként, vagy nem karakterlánc, akkor a metódus új hibát dob.
5. A supportLanguages tulajdonság alapértelmezett értéke egy üres tömb.

### Adja hozzá a programozási nyelv módszerét

Határozzon meg egy addProgrammingLanguage() nevű metódust a CodeEditor osztályban.

1. Van egy addProgrammingLanguage() metódus a CodeEditor osztályban.
2. A metódus egy paramétert kap, amely egy programnyelvi objektum.
3. Az átadott programozási nyelvnek a ProgrammingLanguage osztály példányának kell lennie (ellenkező esetben hibát dob).
4. A metódus hozzáadja a kapott programozási nyelvet az osztály supportLanguages tulajdonságához/tömbéhez.

### Távolítsa el a programozási nyelv módszerét

Határozzon meg egy removeProgrammingLanguage() metódust a CodeEditor osztályban.

1. A CodeEditor osztályban van egy removeProgrammingLanguage() metódus.
2. A metódus egy paramétert kap, amely egy programnyelvi objektum.
3. Az átadott programozási nyelvnek a ProgrammingLanguage osztály példányának kell lennie (ellenkező esetben hibát dob).
4. A metódus eltávolítja a kapott programozási nyelvet az osztály supportLanguages tulajdonságából/tömbjéből.

### OPCIONÁLIS FELADAT:

### Támogatott nyelvek elrejtése

A CodeEditor osztály supportLanguages tulajdonságának elrejtése.

1. A CodeEditor osztály supportLanguages tulajdonsága nem érhető el az objektumon kívülről.

## Tippek

Tesztelési célokra íme néhány példa, amelyeket felhasználhat:

Programozási nyelvek:

* JavaScript (isStaticallyTyped: false, supportsOOP: true)
* Java (StaticallyTyped: igaz, támogatja az OOP-t: igaz)
* C# (isStaticallyTyped: true, supportsOOP: true)

Kódszerkesztők:

* Visual Studio Code (Java, C# és JavaScript támogatása)
* JetBrains IntelliJ IDEA (Java és JavaScript támogatása)
* JetBrains Rider (támogatja a C#-ot és a JavaScriptet)