



Front-end development:

Basics

JavaScript – functions

The following exercise contains the following subjects:

- functions

Instructions

Lets change the code below to use async await instead of .then and .catch

Understanding the code

We are faking an API request to a food recipe database.
We get an array of ids once consumed.

```
const getIDs = () =>
  new Promise((resolve, reject) => {
    setTimeout(() => {
      resolve([532, 543, 753, 1, 5]);
    }, 1500);
  });
```

Here we are fetching a recipe with a given Id.
We are getting a recipe as an object once consumed.

```
const getRecipe = (recipeID) => {  
  return new Promise((resolve, reject) => {  
    setTimeout((ID) => {  
      const recipe = {  
        title: "Fresh tomato pasta",  
        publisher: "Pinchas Hodadad",  
      };  
      resolve(`${ID}: ${recipe.title}`);  
    }, 1500, recipeID);  
  });  
};
```

Here we are consuming the promise.
Getting the Ids, then getting a recipe with the id in the index 2.

```
getIDs()  
  .then((IDs) => {  
    console.log(IDs);  
    return getRecipe(IDs[2]);  
  }).then((recipe) => {  
    console.log(recipe);  
  }).catch((error) => {  
    console.log("It is an error!");  
  });
```

The Task

Your task is to convert the consumed promise not to use .then and .catch but to use the modern way using async and await.