Challenge 3: Account Number of Customers with Balance < $500

Let's test your understanding by solving a slightly tricky challenge.

We'll cover the following

Problem statement

In this challenge, you will be given the names, account numbers, and balances of the customers. Your task is to return the account numbers of customers with a balance of less than $500.

Structure `Account`

To store information about account holders, we have already defined the structure `Account` for you.

```
struct Account {
string name ;      // Stores the name of account holder
int number ;       // Stores the account number of account holder
double balance ;   // Stores the balance of account holder
};
```

Function `subtract`

In this challenge, we have already declared the function `check_account` of type `void` that takes an array of type `Account` and `int`, respectively, in its input

parameters. The function also takes the `size` of the array in its input parameters.

void check_account (struct Account p1[ ] , int arr [ ] , int size)

Initially, `arr` is initialized to **0**. Your task is to find the account number with a balance of less than `$500` and fill the corresponding element in `arr` with a person's account number. If the balance is greater than or equal to **$500**, then fill the corresponding element with **-1**.

You have to write your program logic inside the function `check_account`.

Sample input

```
check_account( {{John ,578328,890.000000 },{Alex ,908210,430.200000 },{Kim ,16
5216,98.500000 }} , {0,0,0} )
```

Sample output

```
{-1,908210,165216}
```

Coding exercise

Before diving directly into the solution, first, try to solve it yourself, and then check if your code passes all the test cases. If you get stuck, you can always see the given solution.

Good Luck! 👍

```
// Structure to store Account information
struct Account {
  string name;
  int number;
  double balance;
};

// Function to find account numbers with less than 500 balance
void check_account(struct Account p[], int arr[], int size) {
  // Write your code here
}
```

🎉Well done! If you have solved the problem, give yourself a round of applause.

✏ Well done! If you have solved the problem, give yourself a round of applause.

In case you are stuck, let's go over the solution review in the next lesson.