

Validating Inputs

Large functions aren't very usable or maintainable. It makes a lot of sense to break down the logic of a program into smaller functions that do one thing each. The program can then call these functions in sequence to perform the work.

Write a program that prompts for a first name, last name, employee ID, and ZIP code. Ensure that the input is valid according to these rules:

- The first name must be filled in.
- The last name must be filled in.
- The first and last names must be at least two characters long.
- An employee ID is in the format AA-1234. So, two letters, a hyphen, and four numbers.
- The ZIP code must be a number.

Display appropriate error messages on incorrect data.

Example Output

```
Enter the first name: J
Enter the last name:
Enter the ZIP code: ABCDE
Enter an employee ID: A12-1234
"J" is not a valid first name. It is too short.
The last name must be filled in.
The ZIP code must be numeric.
A12-1234 is not a valid ID.
```

Or

```
Enter the first name: Jimmy
Enter the last name: James
Enter the ZIP code: 55555
Enter an employee ID: TK-421
There were no errors found.
```

Constraints

- Create a function for each type of validation you need to write. Then create a `validateInput` function that takes in all of the input data and invokes the specific validation functions.
- Use a single output statement to display the outputs.