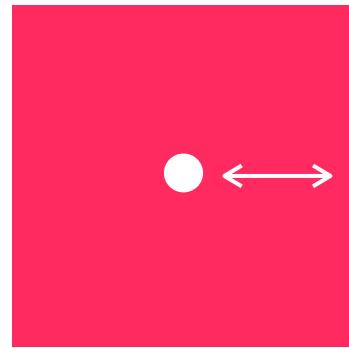
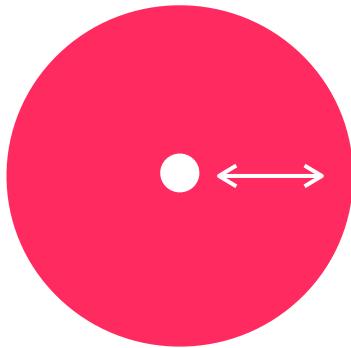


## Senior Backend Developer Test

### Brief

1. This coding test has been designed to assess the coding ability of potential candidates for a Senior Backend Developer role at Spotlas.
2. Once you have completed the test please upload your project on GitHub and email the link to [wasil@spotlas.com](mailto:wasil@spotlas.com) as well as a max 7 minute video explaining and presenting your code and solution.

### Task 1 | Endpoint



Create an endpoint which returns spots in a circle or square area.

Although the task is short, clean code and a good project structure is still very important.  
This task must be completed in Golang.

1. Endpoint should receive 4 parameters
  - ▶ Latitude
  - ▶ Longitude
  - ▶ Radius (in meters)
  - ▶ Type (circle or square)
2. Find all spots in the table (spots.sql) using the received parameters.
3. Order results by distance.
  - ▶ If distance between two spots is smaller than 50m, then order by rating.
4. Endpoint should return an array of objects containing all fields in the data set.

### Resources

- <https://postgis.net>.
- spots.sql

## Task 2 | Comments

**Users can comment on posts and also reply to existing comments.**

1. Prepare two endpoints using Go and PostgreSQL...
  - ▶ GET all comments with their replies.  
    ▶ `type Comment struct { ID int Text string Replies []Comment }`
  - ▶ DELETE a comment (it also should also delete any child comments)
2. There are many different ways to implement these two endpoints. After implementing, justify why your solution is better and what compromises you had to make. Keep in mind that the reply chain can be long.

## Bonus Task | Colouring

1. Prepare a function or pseudocode for changing the background colour of images.
2. Prepare a function or pseudocode for removing small elements from image
  - ▶ Example: Blue lines in left image below.

### Rules and tips:

1. Use only Go and standard libraries.
2. Do not change pixels that are the same colour as background, if they are within a black line or object.
3. If it is easier, you can use 2D array of integers instead of image - where each unique integer represents a different colour.
4. We are not looking for a complete solution, more for an idea of how you could solve the problem.

Before	After
	

*Example image: go.png*