



# Road to Mercari

## Gopher Dojo module 02

*Summary: This document is the subject for the Gopher Dojo module 02 of the Road to Mercari @ 42 Tokyo.*

# Contents

<b>I</b>	<b>Instructions</b>	<b>2</b>
<b>II</b>	<b>Foreword</b>	<b>3</b>
<b>III</b>	<b>Exercise 00 : Typing game</b>	<b>4</b>
<b>IV</b>	<b>Exercise 01 : Concurrent downloads</b>	<b>6</b>

# Chapter I

## Instructions

- Only this page will serve as reference; do not trust rumors.
- Watch out! This document could potentially change up to an hour before submission.
- These exercises are carefully laid out by order of difficulty - from easiest to hardest. We **will not** take into account a successfully completed harder exercise if an easier one is not perfectly functional.
- Make sure you have the appropriate permissions on your files and directories.
- You have to follow the submission procedures for every exercise.
- Your exercises will be checked and graded by your fellow classmates.
- You cannot leave any additional file in your directory than those specified in the subject.
- Got a question? Ask your peer on the right. Otherwise, try your peer on the left.
- Your reference guide is called `Google / man / the Internet / ....`
- Examine the examples thoroughly. They could very well call for details that are not explicitly mentioned in the subject...
- If no other explicit information is displayed, you must assume the following versions of languages : `Go - 1.16.3`.


# Chapter II

## Foreword

<https://gopherdojo.org/>

# Chapter III

## Exercise 00 : Typing game

	Exercise 00
Standard input/output and time limits	
Turn-in directory : <i>ex00/</i>	
Files to turn in : *	

Create a game using standard input/output that meets the following specifications.

- Output a prompt to standard output in alphanumeric characters.
- Accept one line from the user through standard input (also alphanumeric).
- Repeat items 1-2 until the game reaches a time limit of 30 seconds.
- At the end of the time limit, display how many of the prompts the user entered correctly and exit.

Example:

```
?> ./pokemon_quiz
1. Name one type that is super-effective against the Fire type.
-> Water
2. What is the name of the Grass type starter from 1st generation?
-> Bulbasaur
3. How many gym badges do you need to fight the Pokemon League?
-> 8
4. A wild Graveler has appeared! What move will you attack with?
1 - Thunder 2 - Rock Slide
3 - Surf 4 - Flamethrower
->
Time's up! You answered 3 questions correctly.
```




Useful functions for time limit: `time.After`, `context.WithTimeout`



Select statements (wait for time limit and input simultaneously)

# Chapter IV

## Exercise 01 : Concurrent downloads

	Exercise 01
Concurrent downloads	
Turn-in directory : <i>ex01/</i>	
Files to turn in : *	

Create a program that downloads multiple webpages concurrently, meeting the following specifications.

- Accept a variable amount of urls as command-line arguments.

```
?> ./download https://42tokyo.jp/news https://42tokyo.jp/concept/ https://42tokyo.jp/curriculum/
```

- Use the range form of the for loop.
- Each of the urls are downloaded in multiple concurrent goroutines and then merged.
- You must devise a method to handle errors. (use the [golang.org/x/sync/errgroup](https://golang.org/x/sync/errgroup) package)
- You must handle if a cancellation occurs.