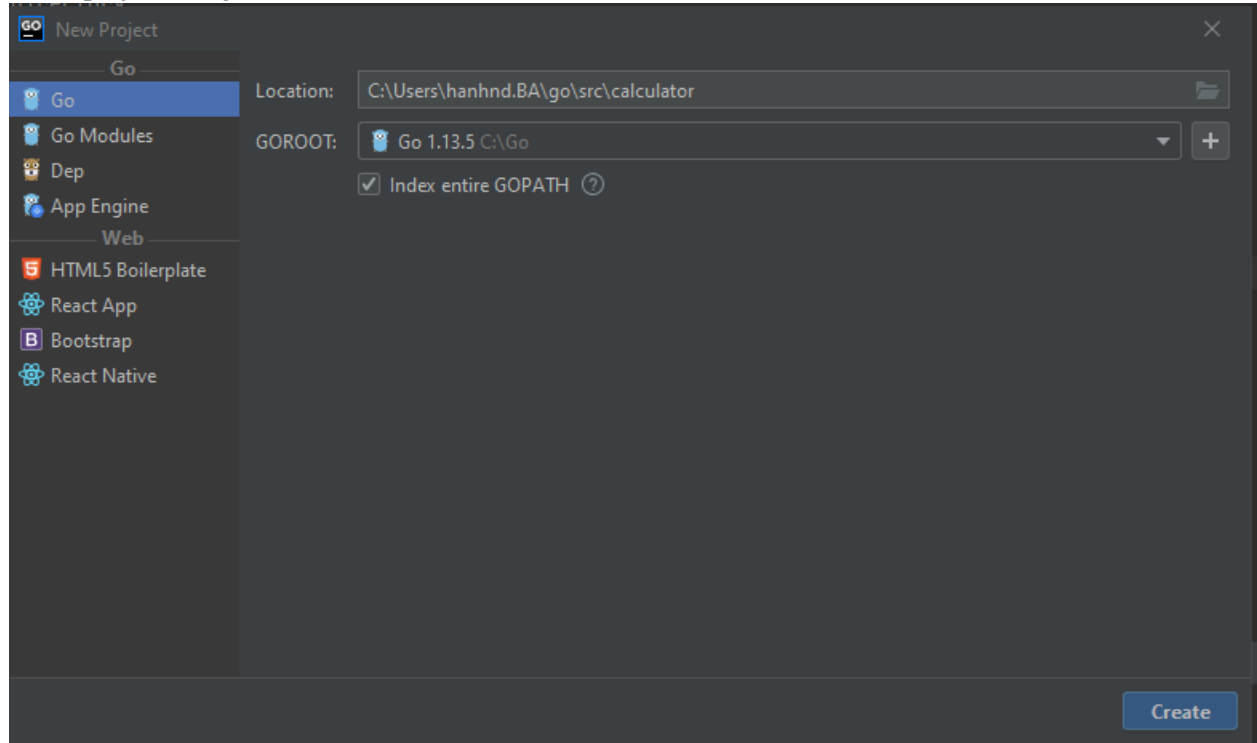


Hướng dẫn tạo một docker image và đẩy lên docker hub

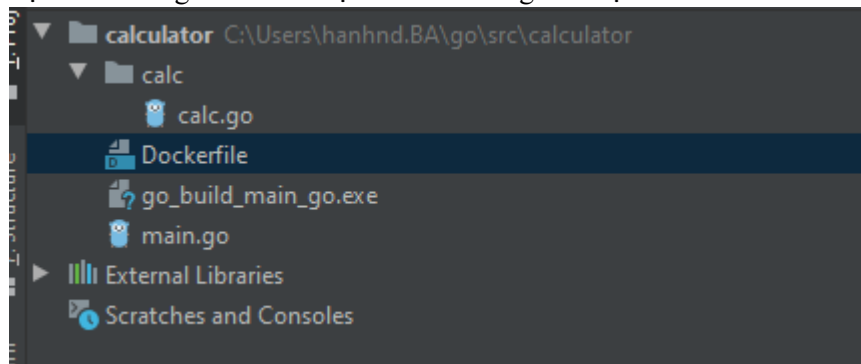
Bài viết sử dụng phần mềm GoLand.

1. Tạo chương trình MyCalculator

- Tạo 1 project trong GoPath có tên là calculator



- Tạo file main.go và 1 thư mục là calc. Trong thư mục calc chứa file calc.go



- File calc.go

```
calc.go x
1  package calc
2  func Add(num1, num2 int) int {
3      return num1 + num2
4  }
5
6  func Subtract(num1, num2 int) int {
7      return num1 - num2
8  }
9
10 func Multiply(num1, num2 float64) float64 {
11     return num1 * num2
12 }
13
14 func Divide(num1, num2 float64) float64 {
15     return num1 / num2
16 }
17
```

- File main.go

- package main

```
import (
    "calculator/calc"
    "fmt"
    "os"
    "strconv"
)

func main() {
    getOperation:
    var operation string
    fmt.Print("Please select an operation: +, -, *, / : ")
    fmt.Scanln(&operation)

    var num1 string
    fmt.Print("Please input the first number: ")
    fmt.Scanln(&num1)

    var num2 string
    fmt.Print("Please input the second number: ")
    fmt.Scanln(&num2)

    switch operation {
    case "+":
        var firstNumber int = strconv.Atoi(num1)
        var secondNumber int = strconv.Atoi(num2)
        fmt.Print("Result: ")
        fmt.Println(calc.Add(firstNumber, secondNumber))
    case "-":
        var firstNumber int = strconv.Atoi(num1)
```

```

    var secondNumber int = stringToInt(num2)
    fmt.Print("Result: ")
    fmt.Println(calc.Subtract(firstNumber, secondNumber))

case "*":
    var firstNumber float64 = stringToFloat64(num1)
    var secondNumber float64 = stringToFloat64(num2)
    fmt.Print("Result: ")
    fmt.Println(calc.Multiply(firstNumber, secondNumber))

case "/":
    var firstNumber float64 = stringToFloat64(num1)
    var secondNumber float64 = stringToFloat64(num2)
    fmt.Print("Result: ")
    fmt.Println(calc.Divide(firstNumber, secondNumber))

default:
    fmt.Println("Invalid operation selected. Please try again!")
    goto getOperation
}
}
func stringToInt(str string) int {
    i, err := strconv.Atoi(str)
    if err != nil {
        fmt.Println(err)
        os.Exit(2)
    }
    return i
}

func stringToFloat64(str string) float64 {
    f, err := strconv.ParseFloat(str, 64)
    if err != nil {
        fmt.Println(err)
        os.Exit(2)
    }
    return f
}
}

```

- Để chạy chương trình ta mở PowerShell, vào thư mục chứa file main.go chạy lệnh  
*go run main.go*

```

PS C:\Users\hanhnd.BA\go\src\calculator> go run main.go
Please select an operation: +, -, *, / : +
Please input the first number: 12
Please input the second number: 14
Result: 26

```

- Để build thành file .exe ta chạy lệnh  
*go build main.go*

## 2. Tạo Dockerfile

- Trong thư mục chứa file main.go tạo file Dockerfile lưu dưới dạng All type trong notepad++

```

- FROM golang:latest
ADD ./go/src/calculator

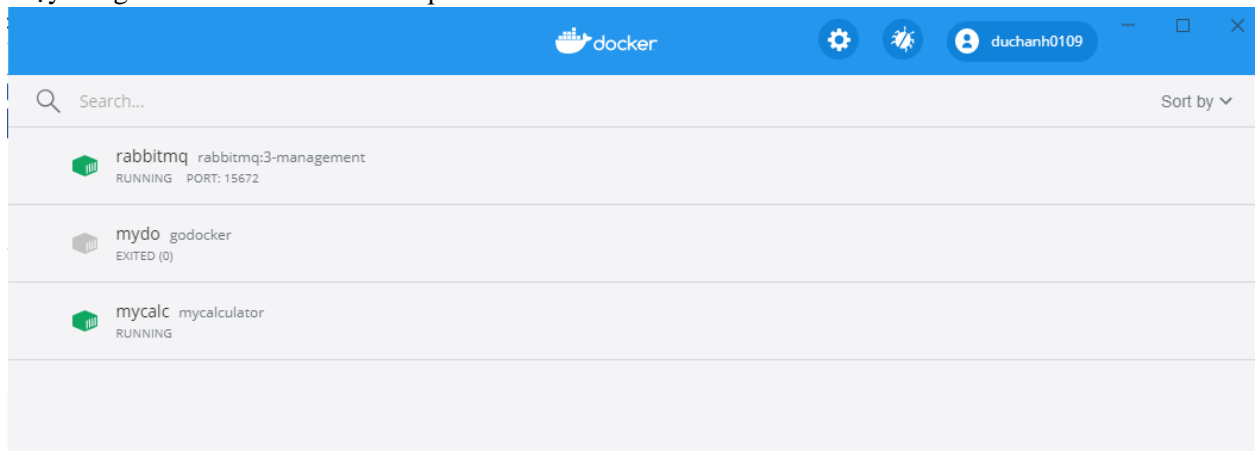
```

```
WORKDIR /go/src/calculator
RUN go install -v ./...
RUN go build -o main .
CMD ["/go/src/calculator"]
```

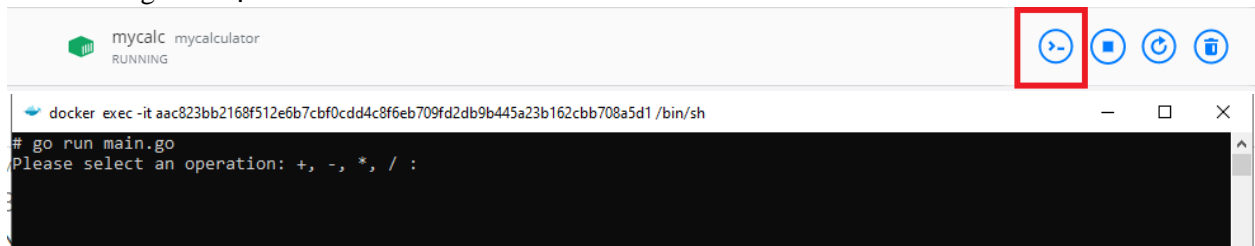
- Mở power shell trong thư mục chứa file main.go
- Chạy lệnh *docker build -t mycalculator*.
  - mycalculator là tên image ta muốn tạo. cuối lệnh là dấu chấm để cho biết thư mục chạy lệnh là thư mục chứa file main.go
- Chờ ít phút để tạo ra image. Để kiểm tra image đã tạo ta chạy lệnh *docker image ls*
- Kết quả

```
PS C:\Users\hanhnd.BA\go\src\calculator> docker image ls
REPOSITORY          TAG         IMAGE ID       CREATED        SIZE
mycalculator         latest     2326f0be3f85   53 seconds ago 816MB
```

- Để chạy image ta vừa tạo ta chạy lệnh *docker run --name mycalc -it mycalculator bash*
- --name là khai báo tên của docker chạy
- bash là để chạy image bằng command
- chạy xong có thể vào docker desktop để kiểm tra



- Để test image vừa tạo



### 3. Push lên docker hub

- Đăng nhập vào docker hub
- Tạo tag cho image mycalculator  
*docker tag mycalculator:latest duchanh0109/mycalc*

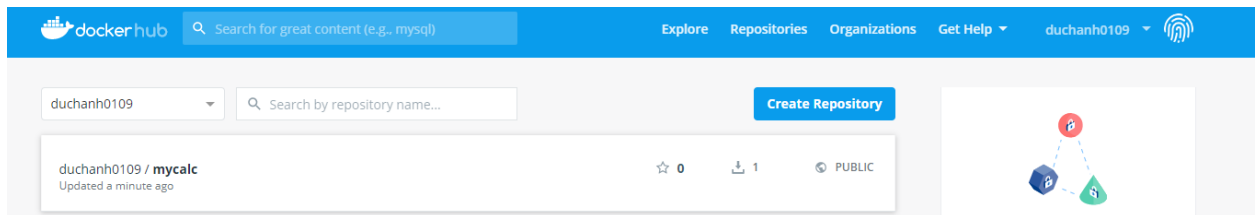
```
PS C:\Users\hanhnd.BA\go\src\calculator> docker tag mycalculator:latest duchanh0109/mycalc
PS C:\Users\hanhnd.BA\go\src\calculator> docker image ls
```

| REPOSITORY         | TAG    | IMAGE ID     | CREATED        | SIZE  |
|--------------------|--------|--------------|----------------|-------|
| mycalculator       | latest | 2326f0be3f85 | 46 minutes ago | 816MB |
| duchanh0109/mycalc | latest | 2326f0be3f85 | 46 minutes ago | 816MB |

- *Đẩy image vừa tag lên docker hub*

```
PS C:\Users\hanhnd.BA\go\src\calculator> docker push duchanh0109/mycalc
The push refers to repository [docker.io/duchanh0109/mycalc]
8fed0f96c628: Pushed
bd916d5da3ce: Pushed
1205405e6ac7: Pushed
a3687a50e862: Mounted from duchanh0109/mygorabbit
dff97196d86d: Mounted from duchanh0109/mygorabbit
2f6bb5712e22: Mounted from duchanh0109/mygorabbit
88cfc2fcd059: Mounted from duchanh0109/mygorabbit
760e8d95cf58: Mounted from duchanh0109/mygorabbit
7cc1c2d7e744: Mounted from duchanh0109/mygorabbit
8c02234b8605: Mounted from duchanh0109/mygorabbit
latest: digest: sha256:4d268009ad114233ef79d7b6f3bb5aae3d881a8ae4c64db7a498bb75ea7ce60b size: 2428
```

- *Kiểm tra trên docker hub*



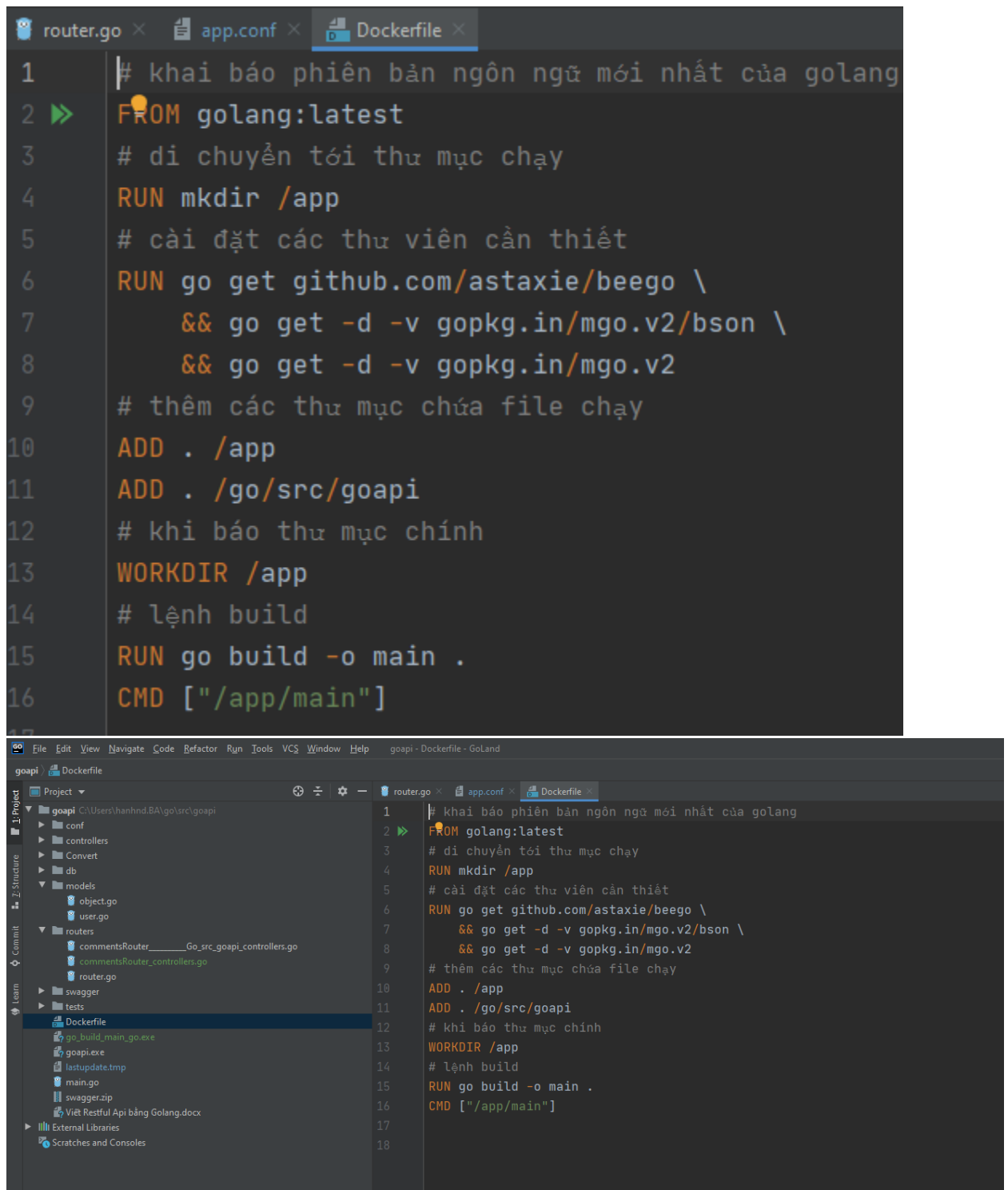
- *Pull thử image vừa đẩy lên về và chạy thử*  
*docker pull duchanh0109/mycalc:latest*  
*docker run --name mytest -it duchanh0109/mycalc bash*

```
PS C:\Users\hanhnd.BA\go\src\calculator> docker pull duchanh0109/mycalc:latest
latest: Pulling from duchanh0109/mycalc
376057ac6fa1: Already exists
5a63a0a859d8: Already exists
496548a8c952: Already exists
2adae3950d4d: Already exists
039b991354af: Already exists
0cca3cbeeb14: Already exists
59c34b3f33f3: Already exists
29a72d33ca44: Pull complete
cc61395c6a0f: Pull complete
de62c43d565a: Pull complete
Digest: sha256:4d268009ad114233ef79d7b6f3bb5aae3d881a8ae4c64db7a498bb75ea7ce60b
Status: Downloaded newer image for duchanh0109/mycalc:latest
duchanh0109/mycalc:latest
PS C:\Users\hanhnd.BA\go\src\calculator> docker image ls
```

| REPOSITORY             | TAG                    | IMAGE ID     | CREATED        | SIZE   |
|------------------------|------------------------|--------------|----------------|--------|
| duchanh0109/mycalc     | latest                 | 2326f0be3f85 | 58 minutes ago | 816MB  |
| godocker               | latest                 | 8b53e7c8ae6f | 3 hours ago    | 852MB  |
| duchanh0109/mygorabbit | 1.0.0.0                | 8b53e7c8ae6f | 3 hours ago    | 852MB  |
| rabbitmq               | 3-management           | 867da7fcd92  | 11 days ago    | 181MB  |
| rabbitmq               | latest                 | 00d1b193f3ff | 11 days ago    | 151MB  |
| golang                 | latest                 | 7e5e8028e8ec | 2 weeks ago    | 810MB  |
| sebp/elk               | latest                 | 253c7beb62ed | 5 weeks ago    | 2.07GB |
| microsoft/dotnet       | 2.2-sdk                | 2357b6790b9d | 5 months ago   | 1.74GB |
| microsoft/dotnet       | 2.2-aspnetcore-runtime | e7e3b238011c | 5 months ago   | 261MB  |

```
PS C:\Users\hanhnd.BA\go\src\calculator> docker run --name mytest -it duchanh0109/mycalc bash
root@8ed6752f1836:/go/src/calculator# go run main.go
Please select an operation: +, -, *, / : -
Please input the first number: 100
Please input the second number: 22
Result: 78
root@8ed6752f1836:/go/src/calculator#
```

#### 4. Ví dụ dockerfile cho beego



The image shows a screenshot of an IDE with two windows. The top window, titled 'Dockerfile', contains the following code:

```
1 | # khai báo phiên bản ngôn ngữ mới nhất của golang
2 | FROM golang:latest
3 | # di chuyển tới thư mục chạy
4 | RUN mkdir /app
5 | # cài đặt các thư viện cần thiết
6 | RUN go get github.com/astaxie/beego \
7 |     && go get -d -v gopkg.in/mgo.v2/bson \
8 |     && go get -d -v gopkg.in/mgo.v2
9 | # thêm các thư mục chứa file chạy
10 | ADD . /app
11 | ADD . /go/src/goapi
12 | # khi báo thư mục chính
13 | WORKDIR /app
14 | # lệnh build
15 | RUN go build -o main .
16 | CMD ["/app/main"]
```

The bottom window shows the project structure of 'goapi' in the IDE. The file tree on the left includes:

- Project
- goapi (C:\Users\hanhnd.BA\go\src\goapi)
  - conf
  - controllers
  - Convert
  - db
  - models
    - object.go
    - user.go
  - routers
    - commentsRouter\_...Go\_src\_goapi\_controllers.go
    - commentsRouter\_controllers.go
    - router.go
  - swagger
  - tests
  - Dockerfile
  - go\_build\_main.go.exe
  - goapi.exe
  - lastupdate.tmp
  - main.go
  - swagger.zip
  - Viết Restful Api bằng Golang.docx
- External Libraries
- Scratches and Consoles

The right pane of the bottom window shows the same Dockerfile code as the top window, with line numbers 1 through 18.

**Dockerfile luôn đặt đồng cấp với file main.go**