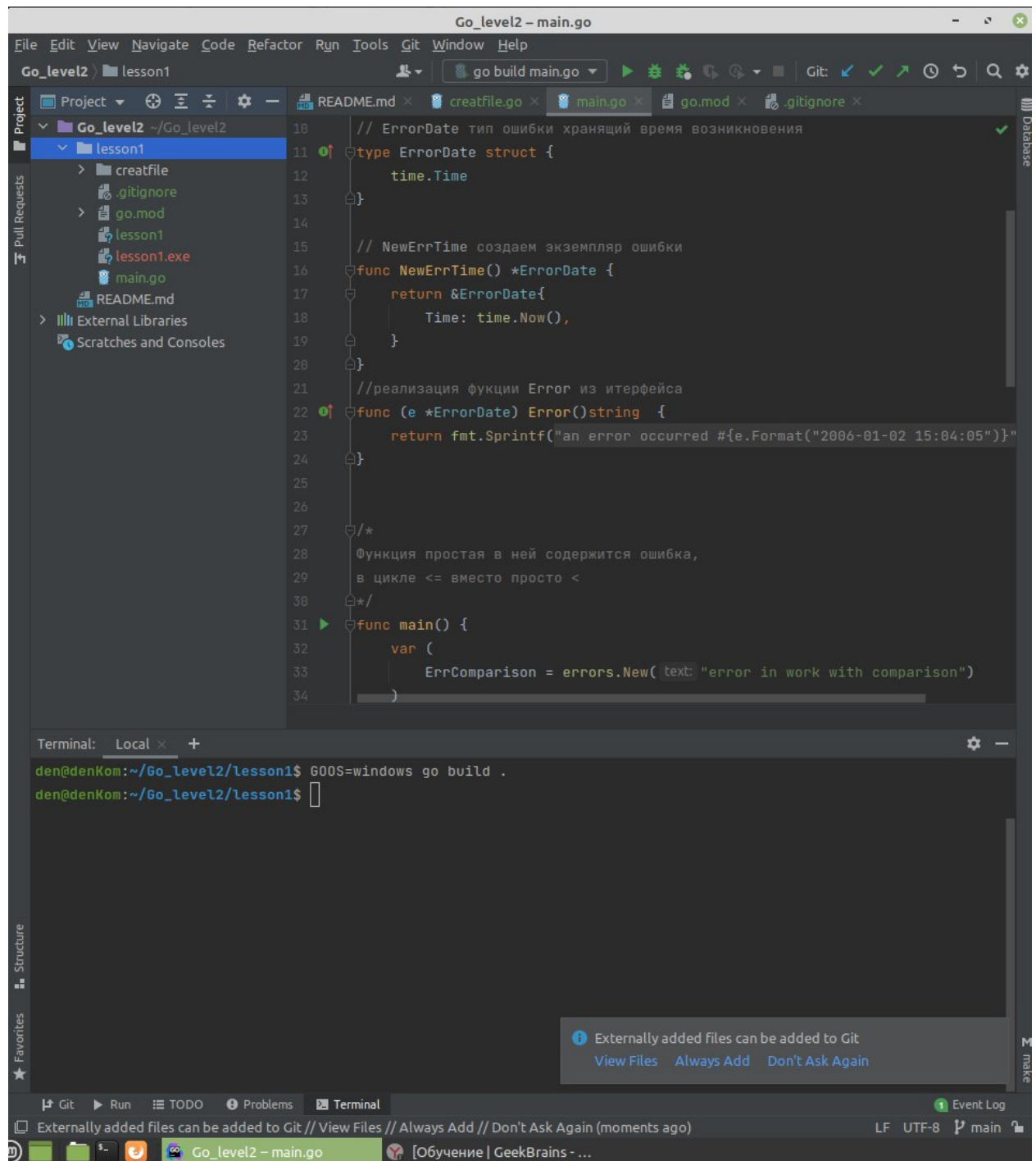


Выполните сборку ваших предыдущих программ под операционную систему, отличающуюся от текущей, скопируйте и запустите ее в виртуальной машине. Пришлите скриншот или фото логов запущенной программы в виртуальной машине. Напишите документацию для одной из предыдущих программ. Создайте документацию, используя подсказки линтера golint, и запустите веб-сервер документации локально. Убедитесь, что документация отображается корректно. Пришлите скриншот отобразившейся документации.

1 Выполняем сборку для OS Windows 10 с помощью команды GOOS=windows go build .



The screenshot shows an IDE window titled "Go\_level2 - main.go". The editor displays Go code for a project named "Go\_level2" with a sub-package "lesson1". The code defines an "ErrorDate" struct with a "time.Time" field, a "NewErrTime" function to create an error instance with the current time, and an "Error" function that formats the error message. The "main" function uses the "errors.New" function to create a new error.

```
10 // ErrorDate тип ошибки хранящий время возникновения
11 type ErrorDate struct {
12     time.Time
13 }
14
15 // NewErrTime создаем экземпляр ошибки
16 func NewErrTime() *ErrorDate {
17     return &ErrorDate{
18         Time: time.Now(),
19     }
20 }
21
22 //реализация функции Error из итерфейса
23 func (e *ErrorDate) Error()string {
24     return fmt.Sprintf("an error occurred #{e.Format("2006-01-02 15:04:05")}"
25 }
26
27 /*
28 Функция простая в ней содержится ошибка,
29 в цикле <= вместо просто <
30 */
31 func main() {
32     var (
33         ErrComparison = errors.New( text: "error in work with comparison")
34     )
```

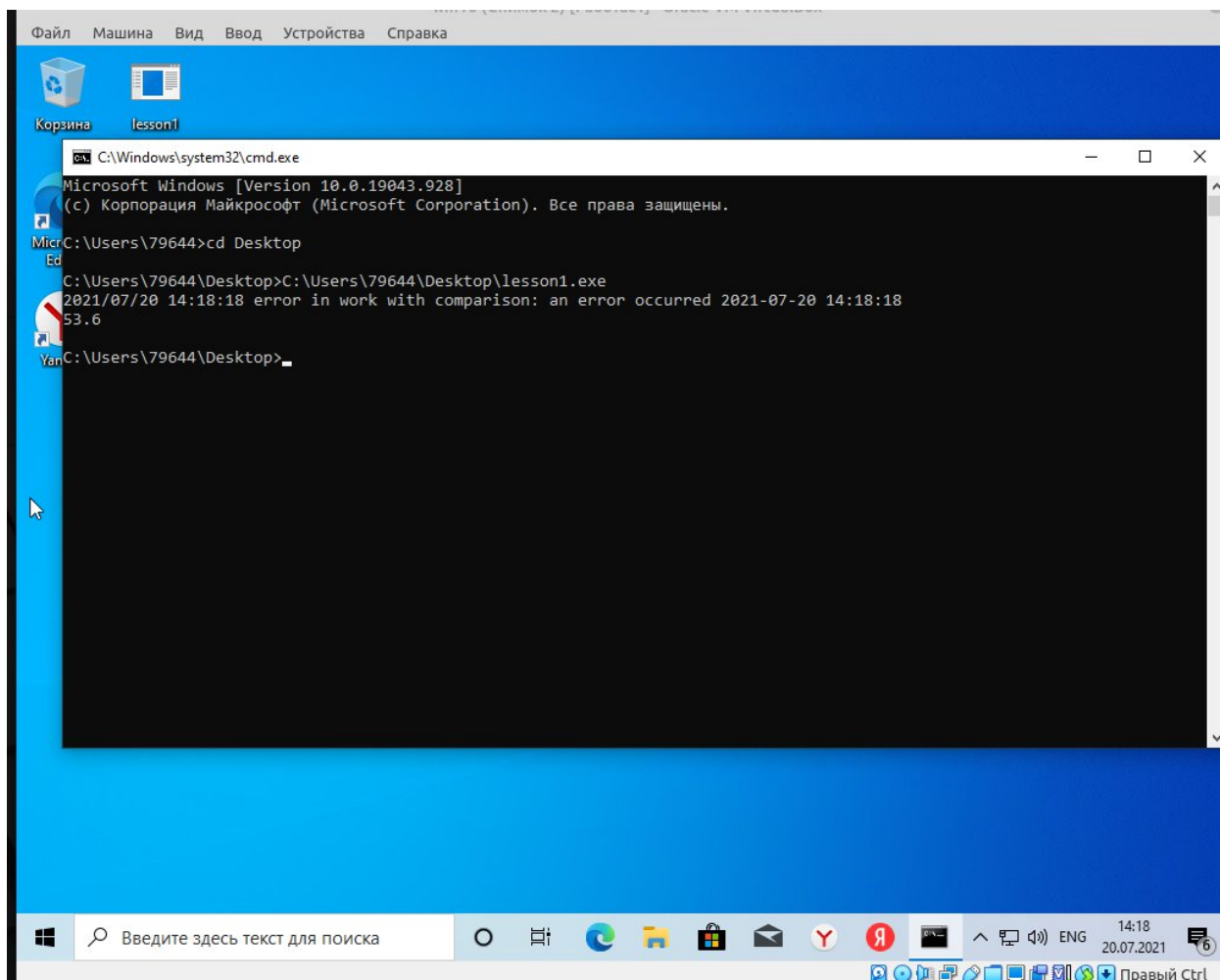
The terminal window at the bottom shows the command "GOOS=windows go build ." being executed in the directory "~/Go\_level2/lesson1". The prompt "den@denKom:~/Go\_level2/lesson1\$" is visible.

Externally added files can be added to Git  
View Files Always Add Don't Ask Again

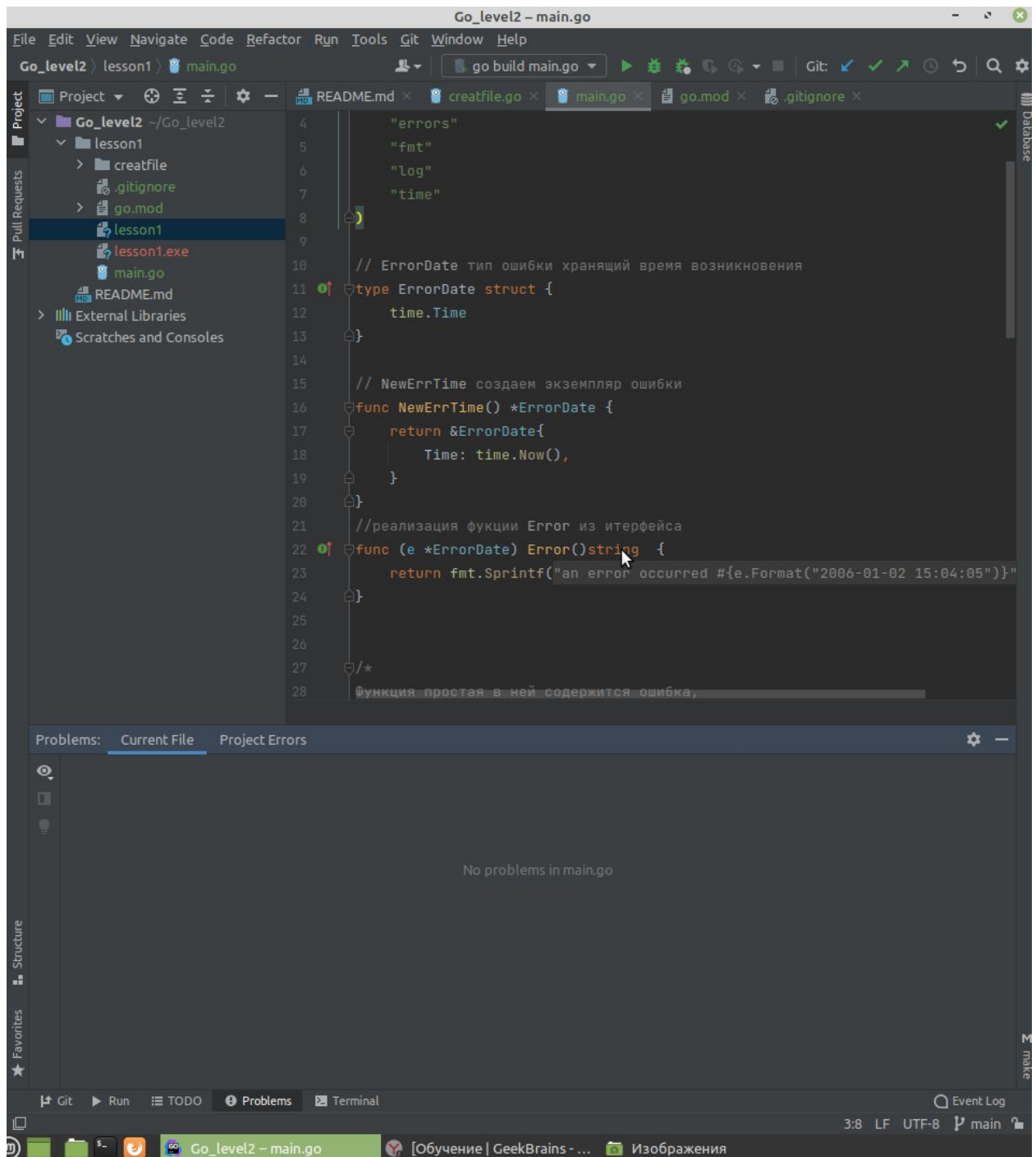
Git Run TODO Problems Terminal Event Log  
Externally added files can be added to Git // View Files // Always Add // Don't Ask Again (moments ago) LF UTF-8 main

Go\_level2 - main.go [Обучение | GeekBrains - ...

## 2 Копируем и запускаем в virtualbox за ранее созданной машиной windows10



## 3 Пишем и создаем документацию, проверяем нет ли ошибок от линтеров



4 Запускаем web server документации командой `godoc -http=:6060`

заходим в браузер на страницу

[http://localhost:6060/pkg/github.com/Go\\_level2/lesson1/](http://localhost:6060/pkg/github.com/Go_level2/lesson1/)

Variables  
type ErrorDate  
func NewErrTime() \*ErrorDate  
func (e \*ErrorDate) Error() string

## Package files

creatfile.go errorDoc.go

## Variables

```
var (  
    ErrNoname = errors.New("error name unknown")  
    ErrCreate = errors.New("it is not possible to create a file ")  
    ErrClose  = errors.New("do not close the file")  
)
```

## type ErrorDate

ErrorDate тип ошибки хранящий время возникновения

```
type ErrorDate struct {  
    time.Time  
}
```

## func NewErrTime

```
func NewErrTime() *ErrorDate
```

NewErrTime создает экземпляр ошибки

## func (\*ErrorDate) Error

```
func (e *ErrorDate) Error() string
```

реализация функции Error из интерфейса