

Что изучать новичку Go lang

Войти на:

<https://git.wildberries.ru> <https://youtrack.wildberries.ru> <https://portal.corp.wildberries.ru/worktime>

Установить:

1. GO <https://golang.org/dl/>
2. Go land <https://www.jetbrains.com/go/promo/> или Visual studio code <https://code.visualstudio.com/>

после установки GO пропишите:

Для Linux

необходимо найти файл .bashrc и добавить в него следующий код: export GOPATH=\$HOME/go и
export PATH=\$PATH:/usr/local/go/bin
в Linux команда: sudo nano имя.файла

Для macOS

необходимо создать файл .bashrc в корневой папке вашего пользователя, и добавить в него следующий код:

```
export GOPATH=$HOME/go  
export GOROOT=/usr/local/opt/go/libexec  
export PATH=$PATH:$GOPATH/bin  
export PATH=$PATH:$GOROOT/bin
```

После чего выполнить source .bashrc для активации настроек.

3. PostgreSQL
<https://www.postgresql.org> <https://losst.ru/ustanovka-postgresql-ubuntu-16-04> <https://eax.me/postgresql-install/>
4. DBeaver для подключение к БД:
<https://dbeaver.io>
5. REST клиент:
<https://insomnia.rest>
6. Для общения пользуемся Telegram.

Ссылки:

- <http://golang-book.ru/>
- <https://go-tour-ru-ru.appspot.com/list>
- <https://gist.github.com/egorsmkv/9df2aef2eddf51986b6d2b5833a4423e> <https://habrahabr.ru/company/mailru/blog/314804/> <https://www.coursera.org/learn/golang-webservices-1/lecture/>

Работа с Git

Научиться пользоваться Git - создать проект, запустить код в свою ветку.
Основы continuous integration, проверка стиля кода и автотестов при отправке данных в gitlab.

Что изучать?

Типы переменных, основной синтаксис.

1. Разобраться в правилах написания json
2. Разобраться со структурами в языке Go. Научиться их создавать, наполнять, парсить json.
3. Разобраться с переменными в Get запросах и Body в POST запросах. Научиться их читать и в дополнение к первому пункту из них наполнять структуры.
4. Разобраться в типах переменных и в том как они передают свои значения. По значению или указателю. В чем разница: uint8, uint16, uint32, uint64, int8, int16, int32 и int64.
5. Разобраться в том что такое горутины и научиться ими пользоваться. Как продолжить разбраться с библиотекой sync, а именно с RWMutex, Wait_Group.
6. Изучить каналы (chan) в go:
в чём разница между буферизированными и не буферизированными
чтениеизканала:вчёмразницамежду v<-с, forv:=ranges, v,ok<-с закрытие канала --
зачем нужно? контрольные вопросы:
что произойдёт при чтении из закрытого канала?
что произойдёт при записи в закрытый канал?
что произойдёт при чтении из неинициализированного канала? что произойдёт при записи в неинициализированный канал?
7. Настроить подключение к базе PSQL из Go.
8. Попробовать создание таблиц в базе из Go (не надо этого делать из Go) Научиться читать и писать в таблицы Postgres запросами напрямую из Go.
9. Изучить что такое хранимые процедуры.
10. Привести не меньше трёх аргументов в пользу того, чтобы доступ к данным в базе из Go шёл именно через хранимки. Это важный пункт!
11. Изучить основы html, css, js.
12. Прочитать про docker, как его собирать, для чего нужен и как с ним работать

13. Почитать про kubernetes, для чего нужен и как с ним работать

Перед тем, как приступить к выполнению тестового задания, советую кратко просмотреть первые 2 лекции:

https://www.youtube.com/watch?v=9Pk7xAT_aCU&list=PLrCZzMib1e9q-X5V9pTM6J0AemRWseM7I