

ЛЕКЦИЯ I © 2020 Нет Ит

JAVA 101:

Git. Debugging

Теодор Костадинов



SOFTWARE
ACADEMY



Съдържание

1. Git
2. GitHub
3. Debugging



Git

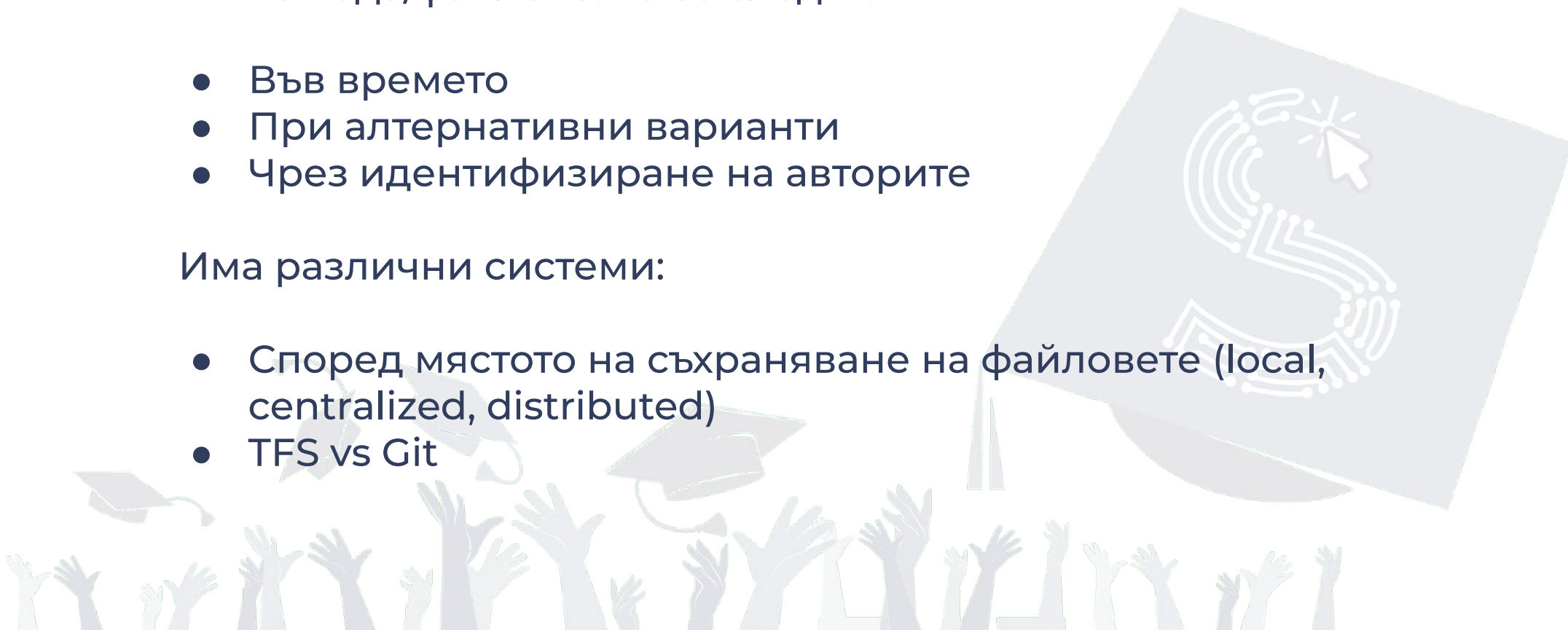
Version Control System

VCS е система, чрез която се следят различните версии на кода/файлове. Те се следят:

- Във времето
- При алтернативни варианти
- Чрез идентифизиране на авторите

Има различни системи:

- Според мястото на съхраняване на файловете (local, centralized, distributed)
- TFS vs Git



Git

- Разработена като страничен проект на създателя на Линукс - Линус Торвалдс през 2005та.
- В нея са решени проблеми, които той и екипът му са срещали при използването на други подобни системи
- Бърза, лесна за използване, напълно дистрибутирана, поддържа огромни проекти
- Позволява не-линейна разработка



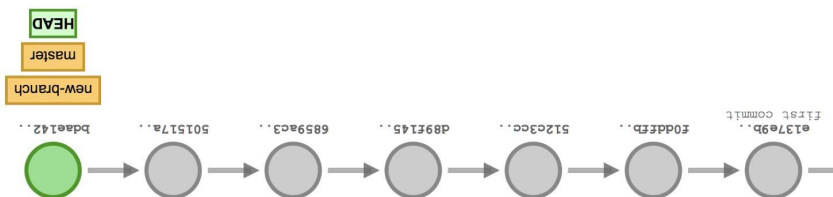
Commits

- Commits са “снимки” на файловете в определен момент
- Всеки commit си има:
 - Автор
 - Дата и час
 - Родител
 - Списък с променените файлове

In case of fire

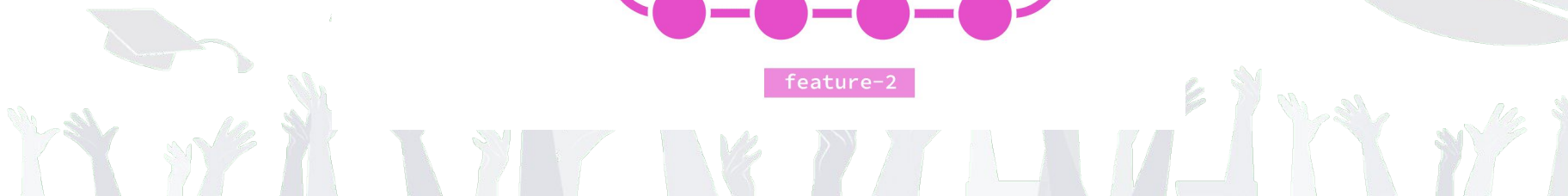
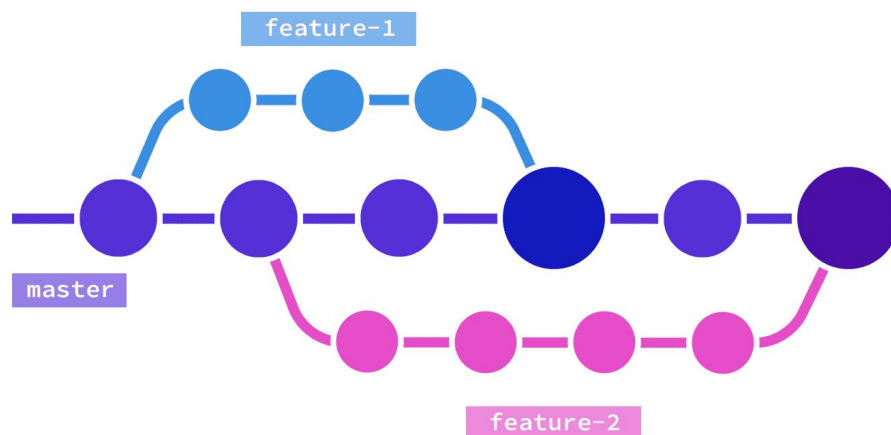


1.  `git commit`
2.  `git push`
3.  `leave building`



Branches

- Гит е изграден на базата на дървовидна структура
- Позволява “разклоняване” и работа по множество “версии” на кода едновременно
- Позволява отделянето или обединяването на версии



Почивка

до 19:50



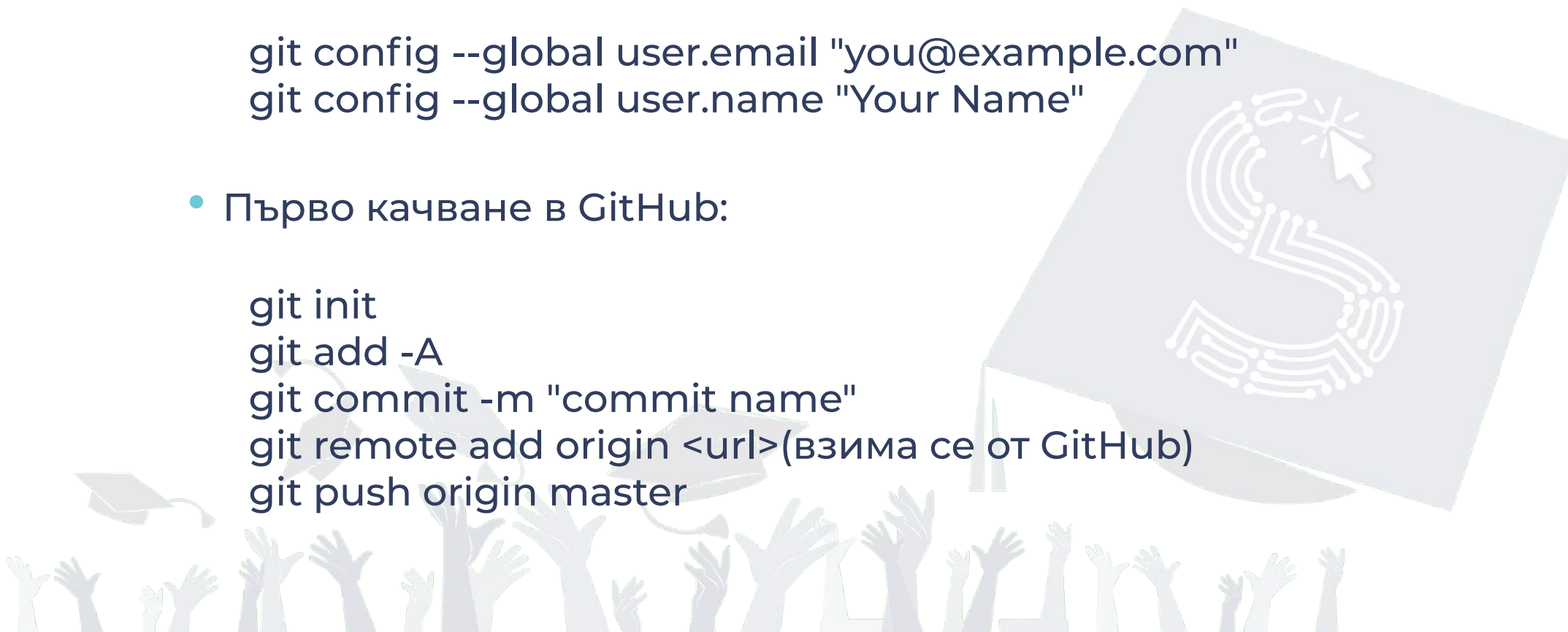
Важни команди

- Команди за първоначален setup:

```
git config --global user.email "you@example.com"  
git config --global user.name "Your Name"
```

- Първо качване в GitHub:

```
git init  
git add -A  
git commit -m "commit name"  
git remote add origin <url>(взима се от GitHub)  
git push origin master
```



Важни команди

- Команди за сваляне на проект:

```
git clone <url>
```

- При качване на кърмити:

```
git add -A (или се изброяват конкретните файлове)
```

```
git commit -m "commit name"
```

```
git pull origin <branch_name> (първо изтегляме  
сървърните промени при нас)
```

```
-----тук оправяме конфликти между нашата и  
сървърната версия, ако има такива-----
```

```
git add -A
```

```
git commit -m "Merge" (правим отделен кърмит за  
конфликтите)
```

```
git push origin <branch_name>
```

GitHub, GitLab, BitBucket, etc..

- Съществуват множество услуги, които предоставят съхранение на кода посредством Гит
- В същността си са еднакви и няма значение коя точно ползвате



Правила

- Къмитите трябва да имат цел:
 - Трябва да се променя само 1 функционалност в къмит
 - Трябва да се именуват правилно и описателно
 - Не трябва да се променят повече от 5-6 файла в къмит

Бранchoвете

- Трябва да следват една и съща конвенция за именуване
- Трябва да имат само свързани помежду си промени



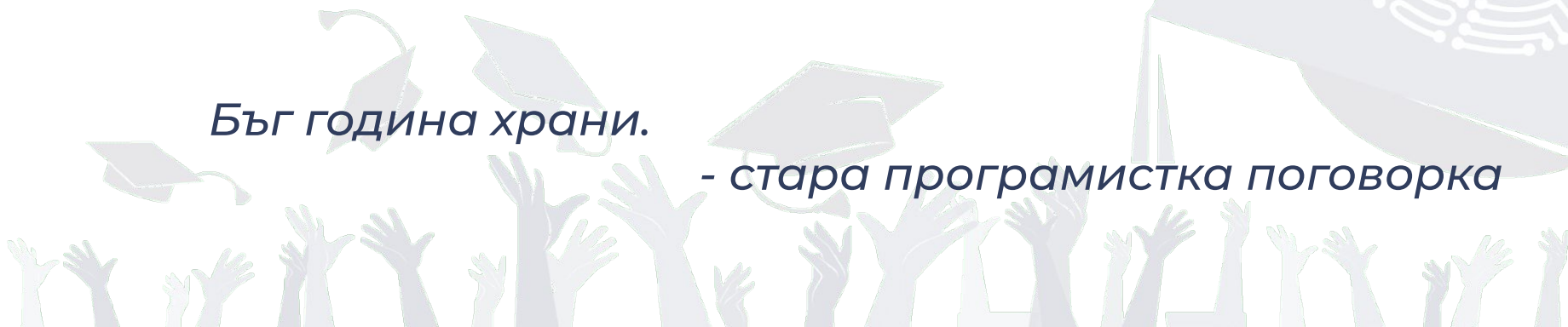
Debugging

Какво е?

- Бъговете са проблеми в кода. Те могат да бъдат:
 - Логически (не сме решили правилно задачата)
 - Технически (знаем как да решим задачата, но не сме успели да го реализираме коректно)
- Дебъгването е процесът по премахване на бъгове

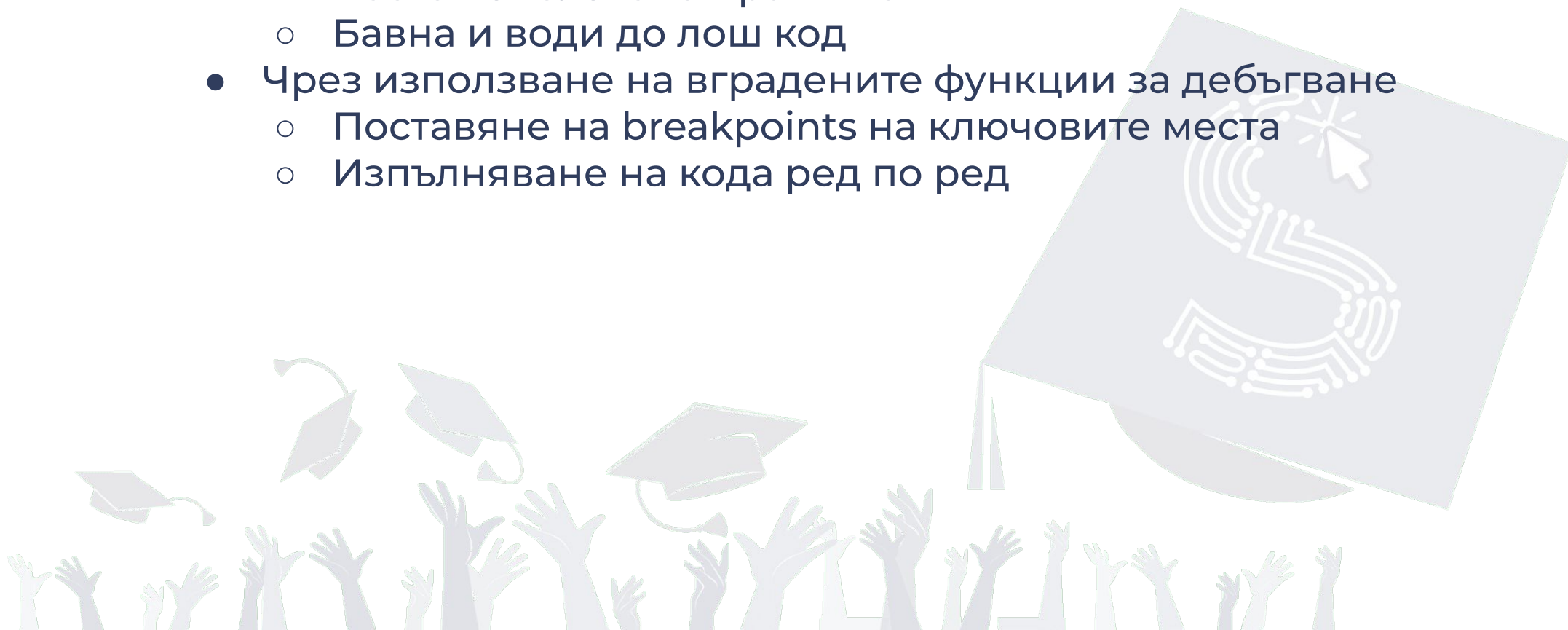
Бъг година храни.

- стара програмистка поговорка



Как се прави?

- Чрез поставяне на логове в кода
 - Често използвана практика
 - Бавна и води до лош код
- Чрез използване на вградените функции за дебъгване
 - Поставяне на breakpoints на ключовите места
 - Изпълняване на кода ред по ред



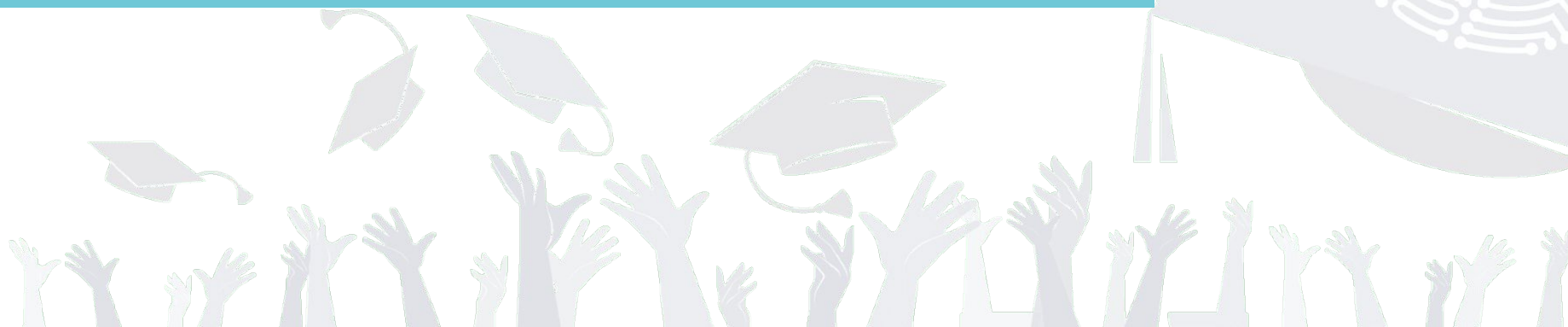
ВЪПРОСИ?



Резюме

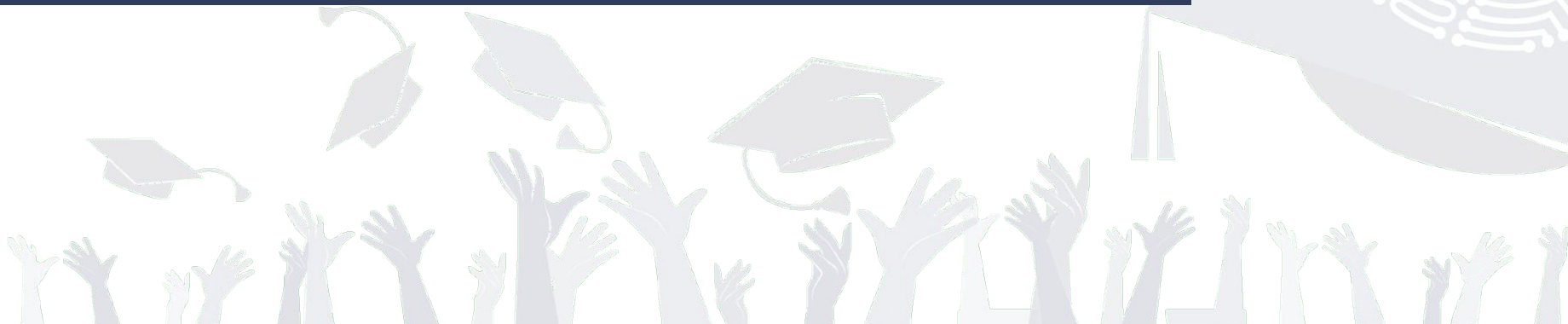


- Гит се използва в реалната насока на проекти и всеки трябва да може да го използва.



Ресурси

- [Debugging Tutorial](#)
-



Задачи за упражняване



© 2020 Нет Ит

БЛАГОДАРЯ ЗА ВНИМАНИЕТО!



SOFTWARE
ACADEMY

