

# Development Tips

CSSE6400

Brae Webb

May 16, 2022

Question

What did we *learn* from the cloud assignment?

Question

What did we *learn* from the cloud assignment?

Answer

Cloud development is *painful*?

Question

How can we make this *less* painful?

Some ideas

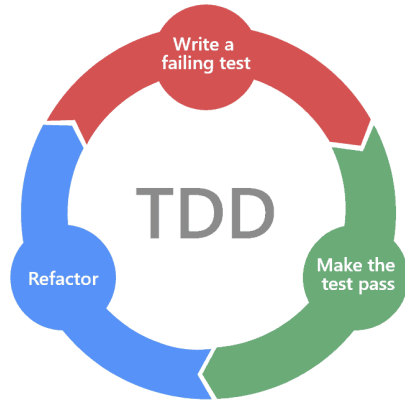
1. Gain confidence through *testing*
2. *Automate* what you're repeating
3. *Tighten feedback loop* as much as possible
4. *Log and monitor* what you can

Gain confidence through *testing*

The testing elephant in the room

# Test-driven development

## TDD in theory



[1]



What do you primarily get from TDD?

1. Regression tests (confidence)

What do you primarily get from TDD?

1. Regression tests (confidence)
2. Small isolated components

Question

Do you have to use TDD to get these benefits?

Question

Do you have to use TDD to get these benefits?

Answer

No.

Question

Do you have to use TDD to get these benefits?

Answer

No. But it can make it easier.

Question

When should you write a test?

Question

When should you write a test?

Answer

Whenever you are checking manually something often

## Examples

- Checking the endpoints your API.



### Examples

- Checking the endpoints your API.
- Checking a feature in the frontend of your interface.

*Automate* what you're repeating

## Examples

- Running tests.

#### Examples

- Running tests.
- Starting locally and running tests locally.

#### Examples

- Running tests.
- Starting locally and running tests locally.
- Deployment (`deploy.sh`).

#### Examples

- Running tests.
- Starting locally and running tests locally.
- Deployment (`deploy.sh`).
- Deployment and running tests.

*Tighten feedback loop* as much as possible

Examples

- Containers



#### Examples

- Containers
- Local database (docker-compose)

#### Examples

- Containers
- Local database (docker-compose)
- Local cloud services (e.g. S3)

*Log and monitor* what you can

Refer to week 9 content

Tip

Enable CloudWatch on EC2

Aside

# Environment variables

## References

[1] Marsner Technologies.

*Why test-driven development (tdd).*

`https://marsner.com/blog/why-test-driven-development-tdd/`.