

3-2 Debugging

In this module you can learn:

- How to react to program errors.
- How to fix them.

Debugging

**What can you do when your
program does not work?**

Debugging in six steps



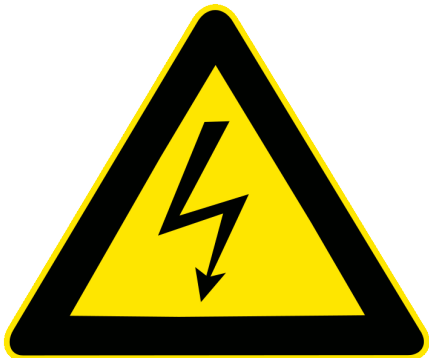
fix syntax errors



**react to runtime
errors**



**look into your
program**



isolate big bugs



**protect against
bugs**



ask for help

Fix Syntax Errors



- Check the line in the error message (and the line before).
- Check for missing colons or brackets.
- Check whether spaces and tabs for indentation are mixed.
- Comment the line. Does the error change?
- Check your Python version (should be 2.7).

React to runtime Errors



- Check the line in the error message.
- Most often: spelling of variable and function names.
- **ImportError**: try import from the shell.
- **IOError**: check file names.
- **ValueError**: check whether you forgot to convert to float or string.
- **IndexError**: usually ore complicated.

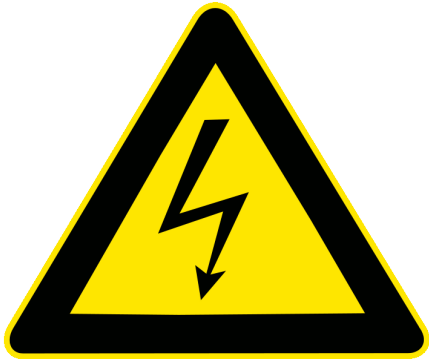
Look into your program



- Add print statements to print variables and check them manually.
- Write intermediate result files.
- Use the Python debugger at the site of the bug:

```
import pdb  
pdb.set_trace()
```

Isolate big bugs



- If the error is complicated, try to narrow it down.
- Use a small set of example data.
- Split your program into functions and test them independently.
- Separate input, process, output in your program.

Protect against bugs



- Don't mix tabs and spaces.
- Use descriptive variable names.
- Avoid redundancy,
e.g. name duplications.
- Avoid **import ***
- Use a good editor.
- Write comments.

Ask for help



- **Debugging is difficult!**
- Ask an experienced programmer.
- Ask a colleague.
- Take a break.
- Explain to your wife/husband/kids what the program should do.
- Read more about programming.

Error types

Match the Python errors with the reasons.

error

`ZeroDivisionError`

`IndexError`

`SyntaxError`

`ValueError`

`IOError`

`ImportError`

possible reason

Adding a string and a number.

A Python module was not found.

A data file was not found.

A list was shorter than expected.

Omitting a bracket or quote.

Dividing by a variable that contained 0.

There is beauty in emptiness

***„Today I had a successful day
programming.
I deleted 300 lines of code.“***