### **NAME**

python-config – output build options for python C/C++ extensions or embedding

### **SYNOPSIS**

```
python-config \ [\ --prefix\ ]\ [\ --exec-prefix\ ]\ [\ --libs\ ]\ [\ --cflags\ ]\ [\ --ldflags\ ]\ [\ --exec-prefix\ ]\ [\ --exec-prefix\ ]\ [\ --help\ ]
```

### DESCRIPTION

**python–config** helps compiling and linking programs, which embed the Python interpreter, or extension modules that can be loaded dynamically (at run time) into the interpreter.

### **OPTIONS**

## --abiflags

print the the ABI flags as specified by PEP 3149.

#### --cflags

print the C compiler flags.

# --ldflags

print the flags that should be passed to the linker.

### --includes

```
similar to --cflags but only with -I options (path to python header files).
```

**--libs** similar to --ldflags but only with -l options (used libraries).

### --prefix

prints the prefix (base directory) under which python can be found.

#### --exec-prefix

print the prefix used for executable program directories (such as bin, sbin, etc).

## --extension-suffix

print the extension suffix used for binary extensions.

--help print the usage message.

#### **EXAMPLES**

To build the singe-file c program prog against the python library, use

```
gcc $(python-config --cflags --ldflags) progr.cpp -o progr.cpp
```

The same in a makefile:

```
CFLAGS+=$(shell python-config --cflags)
LDFLAGS+=$(shell python-config --ldflags)
all: progr
```

To build a dynamically loadable python module, use

```
gcc $(python-config --cflags --ldflags) -shared -fPIC progr.cpp -o progr.so
```

## **SEE ALSO**

```
python (1)
```

http://docs.python.org/extending/extending.html

/usr/share/doc/python/faq/extending.html

### **AUTHORS**

This manual page was written by Johann Felix Soden <johfel@gmx.de> for the Debian project (and may be used by others).