

Devoops Technical Test

If you've applied for our [Sysadmin/devops](#) position, here's our take-home task. You should not need to spend more than two hours on this.

You are to create, on your own and using whatever documentation you can find on a Linux system or on the web, the following three bash scripts.

The scripts are meant to be ran in a Linux system that contains a "base" system with things like `grep`, `sed`, `awk`, `git` and `perl` already installed.

By way of example, such a system might be obtained using Docker:

```
docker run -it --rm debian:buster-slim /bin/bash
apt-get update
apt-get install -yq git-core
```

Docker is *not* a requirement for this test. Any reasonably up-to-date Linux distribution should be sufficient. If you need additional packages that could be found on a stable minimal Debian 10 (Buster) distribution, the scripts will need to check for the command's existence, and return an error noting down a proper installation command for their package if not found.

Here are the three scripts. Note that git is not essential for creating the git hooks.

1. A bash-based pre-receive Git hook.

If more than one reference has been pushed, the script must report the names of the pushed references and then abort the push. Otherwise, the script must not produce any output.

2. A bash-based post-receive Git hook

This script must check all pushed references and:

- Output "Deploying to live" if the name of the reference is "refs/heads/master".

- Output "Deploying to staging" if the name of the reference is "refs/heads/develop".

3. A bash-based stand-alone script

Create a stand-alone script called `suffix-search` that searches a directory tree for files that match one or more file basename specifications and which have a modification time within a given range (inclusive of start and end times). The script must accept the following arguments in an order that you decide:

- A required path that specifies the root of the directory tree that is to be searched.
- A required user-specified list of file base name specifications (pathnames without directories). The filenames must support `?`, `*`, and `[...]` wildcards (glob patterns). At least one must be provided by the user.
- A required date/time *range* in [Unix time](#) format (the number of seconds that have elapsed since the Unix epoch, 00:00:00 UTC on 1 January 1970).
- An optional `--help` argument that causes the script to display appropriate help text.

The script must produce an appropriate exit code so that it can easily be used as part of a pipeline that determines if a directory tree contains files that match the supplied specification:

```
$ suffix-search [...] && echo 'found' || echo 'not found'
```

ABOUT US

MORE CONTENT

SITUATION PUBLISHING



SITUATION
PUBLISHING

The Register - Independent news and views for the
tech community. Part of Situation Publishing

SIGN UP TO OUR DAILY NEWSLETTER

Your Work Email Address

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

SUBSCRIBE

Biting the hand that feeds IT © 1998–2020

[Your Consent Options](#)

[Cookies](#)

[Privacy](#)

[Ts&Cs](#)