

【Linux Programming】 Day8

☰ Tags	
📅 Date	@May 28, 2022
☰ Summary	Command :, continue, and break

2.5 Commands

2.5.1 break

Use `break` for escaping from an enclosing `for`, `while`, or `until` loop before the controlling condition has been met.

We can also give `break` an additional numeric parameter, which is the number of loops to break out of.

```
#!/bin/sh

rm -rf fred*
echo > fred1
echo > fred2
mkdir fred3
echo > fred4

for file in fred*
do
    if [ -d "file" ]; then
        break;
    fi
done

echo "First directory starting fred was $file"

exit 0
```

2.5.2 The : Command

The `colon command` is a `null` command. It's occasionally useful to simplify the logic of conditions, being an alias for `true`. `while :` implements an infinite loop in place of the morcommon `while true`

```
#!/bin/sh

rm -f fred

if [ -f fred ]; then
:
else
    echo "File fred does not exist"
fi

exit 0
```

2.5.3 continue

Rather like the C statement, this commands makes the enclosing for, while, or until loop continue at the next iteration.

```
#!/bin/sh

rm -rf fred*
echo > fred2
mkdir fred3
echo > fred4

for file in fred*
do
    if [ -d "$file" ]; then
        echo "Skipping directory file $file"
    fi
    echo "File is $file"
done

rm -rf fred*
exit 0
```