

Labwork 9 – AWK Prep & Worked Examples

Quick practice tasks with embedded solutions so you can warm up before the graded lab. Each example uses the same patterns as the lab: CSV-like fields, date parsing, prefix checks, and grouped counts.

Run scripts with `awk -f <script>.awk <input>`. All scripts include the `#!/bin/awk -f` shebang so you can `chmod +x` and execute directly.

Example 1 — Logins (year filter)

`prep_std.txt`:

```
std01,Ada,Lovelace,12/06/2017,09:10
ins02,Grace,Hopper,13/06/2017,10:00
std03,Alan,Turing,01/01/2018,08:30
ins04,Edsger,Dijkstra,15/12/2017,11:45
```

`prep_logins.awk`:

```
#!/bin/awk -f
BEGIN { FS="," }
{
    split($4, d, "/")           # d[1]=day, d[2]=month, d[3]=year
    if (d[3] != "2017") next

    if (index($1, "std") == 1) students++
    else if (index($1, "ins") == 1) instructors++
}
END {
    print "Students:", (students ? students : 0)
    print "Instructors:", (instructors ? instructors : 0)
}
```

Expected output:

```
Students: 1
Instructors: 2
```

Example 2 — Library (month/year + type)

`prep_library.txt`:

FIC111,Foundation,Sci-Fi,02/03/2024,STD
NON222,AI: A Modern Approach,Education,05/03/2024,FAC
FIC333,Dune,Sci-Fi,10/02/2024,STD
NON444,Linear Algebra,Education,15/03/2024,STD

prep_library.awk:

```
#!/bin/awk -f
BEGIN { FS="," }
{
    split($4, d, "/")          # d[2]=month, d[3]=year
    if (d[2] != "03" || d[3] != "2024") next

    isFiction = (index($1, "FIC") == 1)
    isStudent = ($5 == "STD")

    if (isStudent) {
        if (isFiction) std_fic++; else std_non++;
    } else {
        # assume remaining are faculty
        if (isFiction) fac_fic++; else fac_non++;
    }
}
END {
    print "Students borrowed:"
    print "Fiction Books:", (std_fic ? std_fic : 0)
    print "Non-Fiction Books:", (std_non ? std_non : 0)
    print "Faculty borrowed:"
    print "Fiction Books:", (fac_fic ? fac_fic : 0)
    print "Non-Fiction Books:", (fac_non ? fac_non : 0)
}
```

Expected output:

```
Students borrowed:
Fiction Books: 1
Non-Fiction Books: 1
Faculty borrowed:
Fiction Books: 1
Non-Fiction Books: 0
```

Example 3 — Gym Activities (month/year + category)

prep_gym.txt:

```
STD500,Sam,Basic,03/02/2024,GYM
PR0600,Lou,Premium,05/02/2024,YOGA
STD700,Tay,Basic,06/02/2024,YOGA
PR0800,Max,Premium,10/01/2024,GYM
STD900,Rae,Basic,20/02/2024,SWIM
```

prep_gym.awk:

```
#!/bin/awk -f
BEGIN { FS="," }
{
    split($4, d, "/")
    if (d[2] != "02" || d[3] != "2024") next

    isStudent = (index($1, "STD") == 1)

    if (isStudent) {
        student[$5]++
    } else {
        professional[$5]++
    }
}
END {
    printf "Student Activities:\n"
    print "GYM:", (student["GYM"] ? student["GYM"] : 0)
    print "YOGA:", (student["YOGA"] ? student["YOGA"] : 0)
    print "SWIM:", (student["SWIM"] ? student["SWIM"] : 0)

    printf "Professional Activities:\n"
    print "GYM:", (professional["GYM"] ? professional["GYM"] : 0)
    print "YOGA:", (professional["YOGA"] ? professional["YOGA"] : 0)
    print "SWIM:", (professional["SWIM"] ? professional["SWIM"] : 0)
}
```

Expected output:

```
Student Activities:
GYM: 1
YOGA: 1
SWIM: 1
Professional Activities:
GYM: 0
YOGA: 1
SWIM: 0
```

- Type in the tiny datasets, run the scripts, and confirm you match the expected output.
- Then tweak the inputs (add lines, change months/years) to see the filters drop or include rows.
- Once comfortable, build the graded scripts by copying the skeletons and adjusting field names and categories to the lab specs.