Regular Expressions: Code Snippets

Input string

```
The man was running down the street.
The man was looking for some food to eat.
```

1. Exact match

```
man
```

2. Any character

```
e.t
```

3. One or more

```
e+t
```

4. Zero or more

```
ee*t
```

5. Optional/If

```
ea?t.
```

6. Min,Max

```
e{1,2}t
```

7. Escape Character/ Special Character

```
\.
```

8. Set of characters

```
[a-z]*ing
```

9. Group

```
r(ee|nn)t
```

10. Sepcials

```
\w
\W
\s
\S
\d
```

Validate a Phone Number

Valid Inputs

```
1234567890
123-456-7890
(123) 456 7890
123 456 7890
```

2. Solution

```
\(?[0-9]{3}\)?(-| )?[0-9]{3}(-| )?[0-9]{4}
OR
\(?\d{3}\)?(-| )?\d{3}(-| )?\d{4}
```

Validate an Email Address

Conditions

```
* Start with a lowercase alphabet
* Contains at least 4 characters before @
* Valid character set -> a-z, 0-9 and _
* Should contain @
* Should end in .com
* domain should not be empty
```

Example

```
test@test.com -> Valid
123@test.com -> Invalid
test@test.in -> Valid
_1234@test.com -> Invalid
12@test.com -> Invalid
test@.com -> Invalid
```

Solution

```
^[a-z][a-z0-9_][a-z0-9_][a-z0-9_]+@[a-z0-9_]+\.com
```

AWK: Code Snippets

Input Data

```
Max 10 100
Sam 5 50
Beth 10 0
John 20 100
Tim 5 50
```

1. {Pattern Action} Syntax

```
awk '$3 > 0 {print $1}' emp.txt
```

2. Action Syntax

```
awk '{print $0}' emp.txt
```

3. Basic MATH

```
awk '{print $1, $2*$3}' emp.txt
```

4. REGEX

```
awk  ' /.a.*/ {print $1, $2*$3}' emp.txt
```

5. REGEX on a column

```
awk  ' $1 ~ /.a.*/ {print $1, $2*$3}' emp.txt
```

6. AND Operator

```
awk  ' $2>3 && $3<60 {print $1, $2*$3}' emp.txt
```

7. Special Variables

```
awk  '{print NF, NR}' emp.txt
```

8. BEGIN PROCESS END Syntax

```
awk 'BEGIN {print("Do Initializatiosn here")} {print $0} END{ print("Do Output Formatting here")}' emp.txt
```