MIT-WPU

।। विश्वशान्तिर्ध्रुवं ध्रुवा ।।

Dr. Vishwanath Karad

MIT WORLD PEACE UNIVERSITY | PUNE

TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

School of Computer Engineering and Technology

Lab Assignment-09

Create a Regular Expression and implement the following

a) Recognize the following strings: “bat,” “bit,” “but,” “hat,” “hit,” or “hut.”

b) Match any pair of words separated by a single space, i.e., first and last names.

c) Match any word and single letter separated by a comma and single space, as in last name, first initial.

Regular Expression

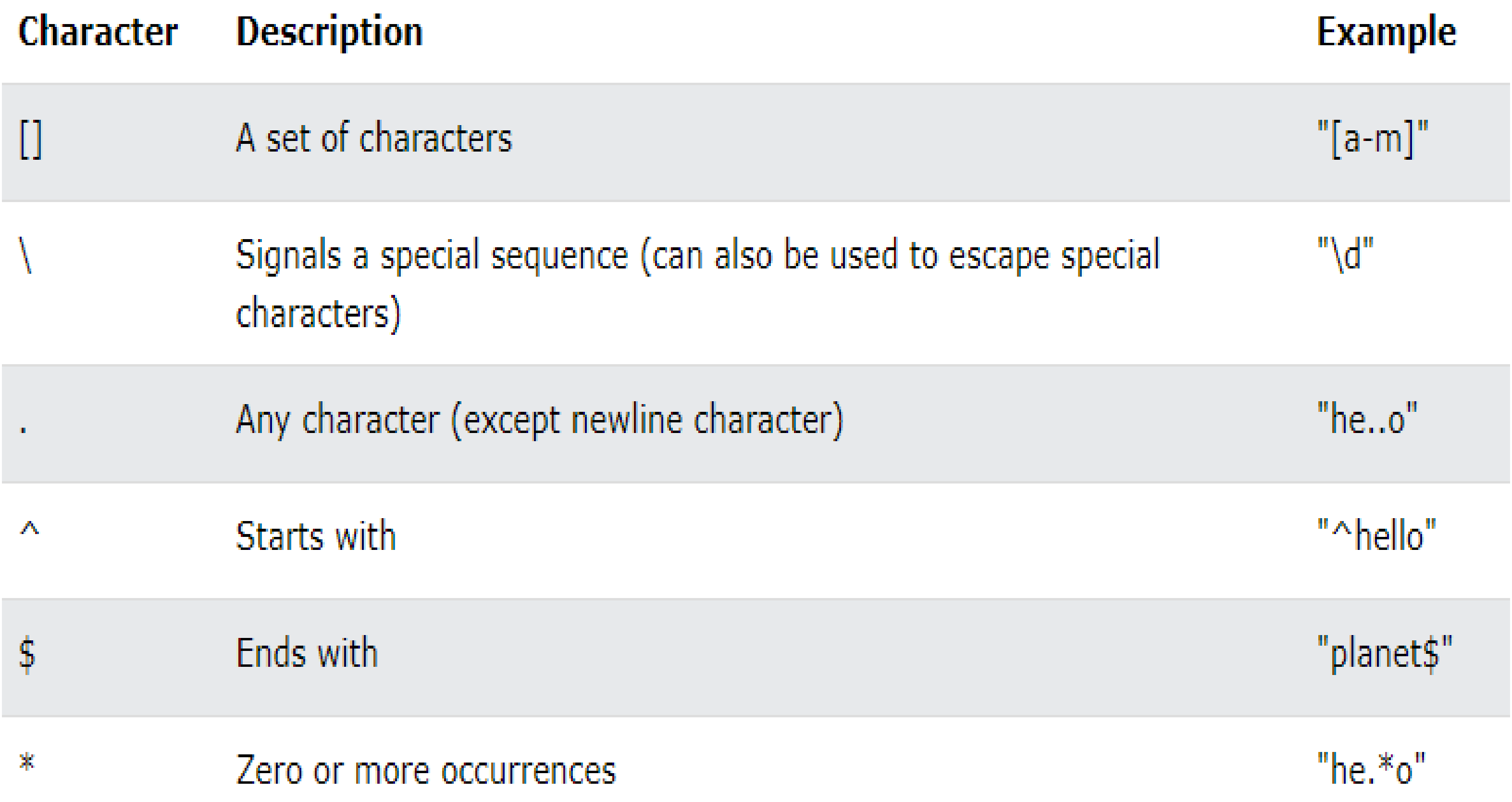
- A RegEx, or Regular Expression, is a sequence of characters that forms a search pattern.

- RegEx can be used to check if a string contains the specified search pattern.

- Python has a built-in package called re, which can be used to work with Regular Expressions.
- import re

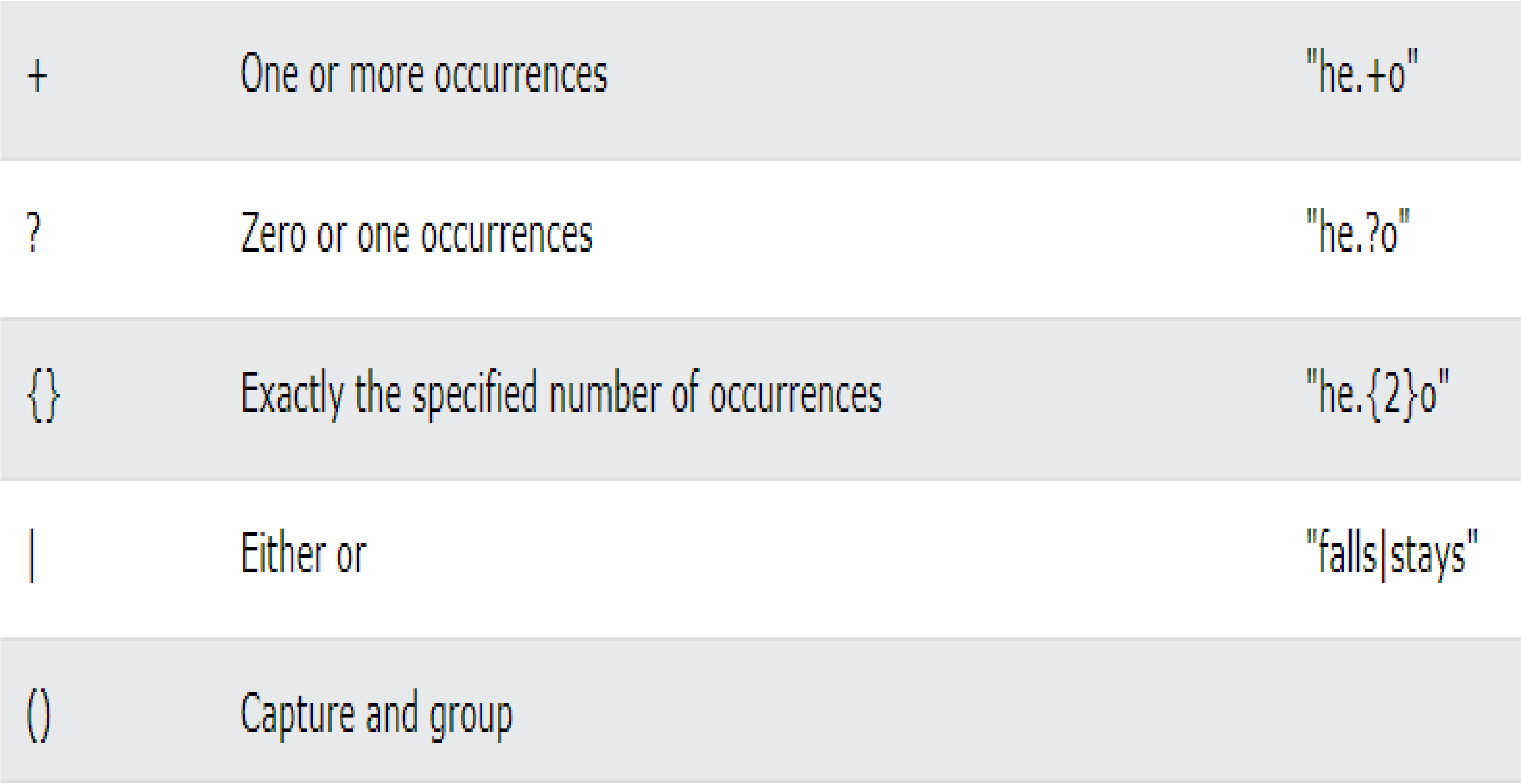
Metacharacters

Metacharacters are characters with a special meaning:



Character	Description	Example
[]	A set of characters	"[a-m]"
\	Signals a special sequence (can also be used to escape special characters)	"\d"
.	Any character (except newline character)	"he..o"
^	Starts with	"^hello"
$	Ends with	"planet$"
*	Zero or more occurrences	"he.*o"

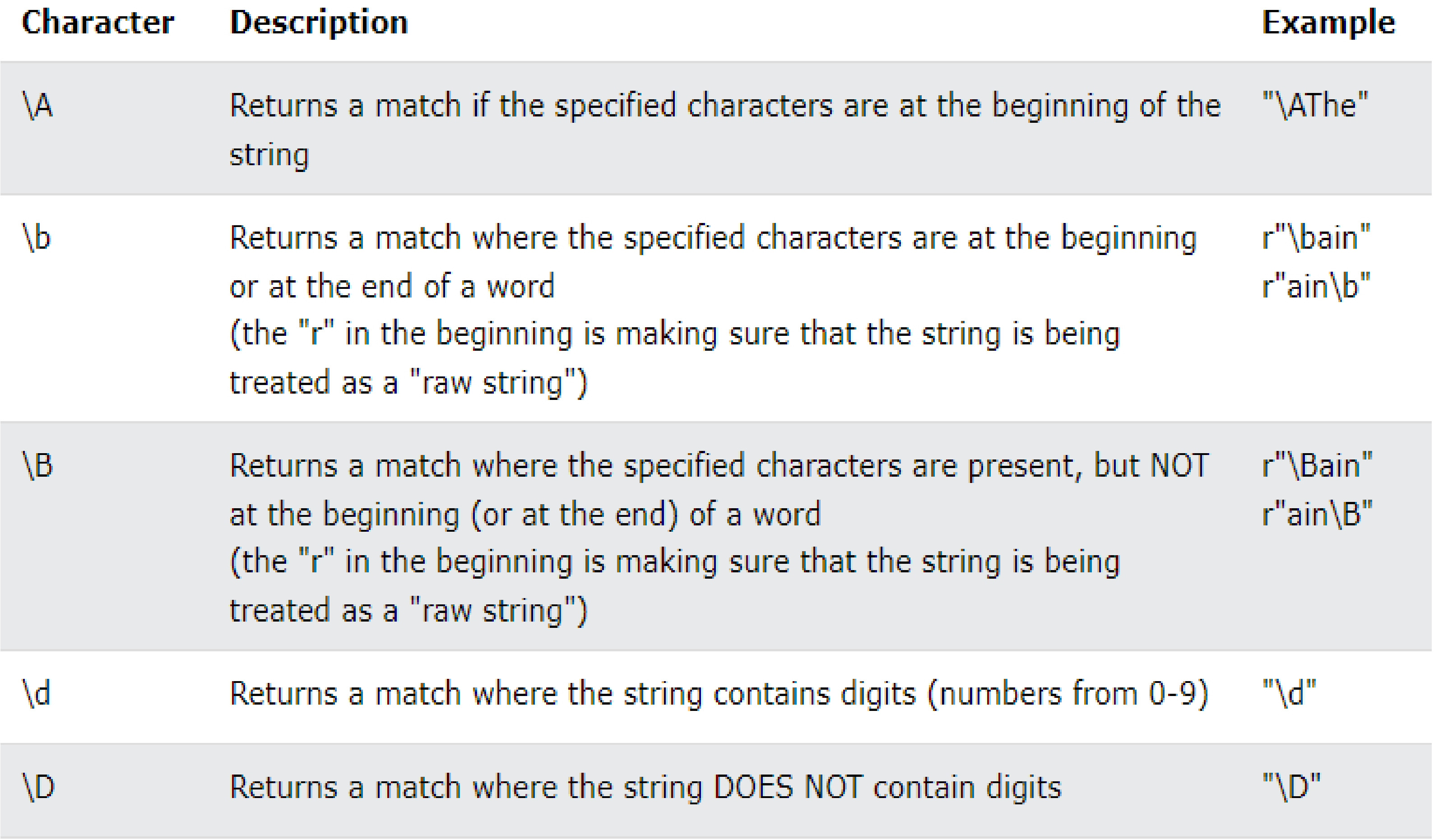
Metacharacters



+	One or more occurrences	"he.+o"
?	Zero or one occurrences	"he.?o"
{}	Exactly the specified number of occurrences	"he.{2}o"
|	Either or	"falls|stays"
()	Capture and group	

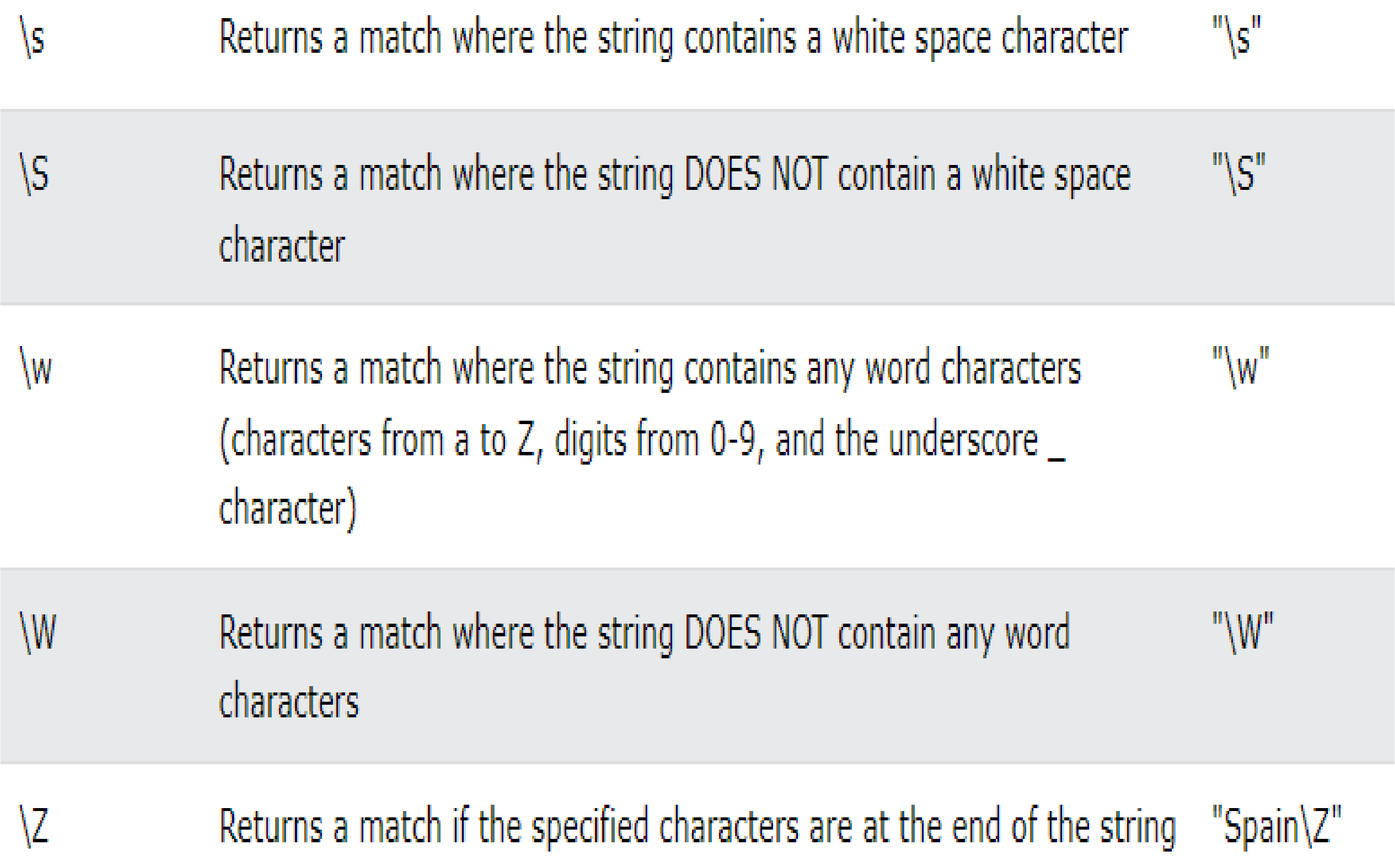
Special Sequences

A special sequence is a \ followed by one of the characters in the list below, and has a special meaning:



Character	Description	Example
\A	Returns a match if the specified characters are at the beginning of the string	"\AThe"
\b	Returns a match where the specified characters are at the beginning or at the end of a word (the "r" in the beginning is making sure that the string is being treated as a "raw string")	r"\bain" r"ain\b"
\B	Returns a match where the specified characters are present, but NOT at the beginning (or at the end) of a word (the "r" in the beginning is making sure that the string is being treated as a "raw string")	r"\Bain" r"ain\B"
\d	Returns a match where the string contains digits (numbers from 0-9)	"\d"
\D	Returns a match where the string DOES NOT contain digits	"\D"

Special Sequences



\s	Returns a match where the string contains a white space character	"\s"
\S	Returns a match where the string DOES NOT contain a white space character	"\S"
\w	Returns a match where the string contains any word characters (characters from a to Z, digits from 0-9, and the underscore _ character)	"\w"
\W	Returns a match where the string DOES NOT contain any word characters	"\W"
\Z	Returns a match if the specified characters are at the end of the string	"Spain\Z"

RegEx Functions

The re module offers a set of functions that allows us to search a string for a match:

Function	Description
findall	Returns a list containing all matches
search	Returns a Match object if there is a match anywhere in the string
split	Returns a list where the string has been split at each match
sub	Replaces one or many matches with a string

The findall() Function

The findall() function returns a list containing all matches.

If no matches are found, an empty list is returned.

E.g

```
import re
txt = "The rain in Spain"
x = re.findall("ai", txt)
print(x)
```

Output:

```
['ai', 'ai']
```

The search() Function

The search() function searches the string for a match, and returns a Match object if there is a match.

Eg.

Search for the first white-space character in the string:

```
import re
txt = "The rain in Spain"
x = re.search("\s", txt)
print("The first white-space character is located in position:", x.start())
```

Output:

```
The first white-space character is located in position: 3
```

The split() Function

The split() function returns a list where the string has been split at each match:

Eg.

Split at each white-space character:

```
import re

txt = "The rain in Spain"
x = re.split("\s", txt)
print(x)
```

Output:

```
['The', 'rain', 'in', 'Spain']
```

The sub() Function

The sub() function replaces the matches with the text of your choice:

Eg.

Replace every white-space character with the number 9:

```
import re

txt = "The rain in Spain"
x = re.sub("\s", "9", txt)
print(x)
```

Output:

```
The9rain9in9Spain
```

Thank you