

# Easy Test Analysis

## Introduction

The easy test focuses on basic XML parsing using the xml2 package. It involves extracting specific information from a simple XML document. The code snippet below demonstrates how to load the xml2 package and parse a simple XML document to extract the director name for the second movie.

## Setting Up the Environment

### Section 1: Loading Libraries and XML String

```
library(xml2)
library(stringr)

xml_string <- c( '<?xml version="1.0" encoding="UTF-8"?>',
  '<movies>',
  '<movie mins="126" lang="eng">',
  '<title>Good Will Hunting</title>',
  '<director>',
  '<first_name>Gus</first_name>',
  '<last_name>Van Sant</last_name>',
  '</director>',
  '<year>1998</year>',
  '<genre>drama</genre>',
  '</movie>',
  '<movie mins="106" lang="spa">',
  '<title>Y tu mama tambien</title>',
  '<director>',
  '<first_name>Alfonso</first_name>',
  '<last_name>Cuaron</last_name>',
  '</director>',
  '<year>2001</year>',
  '<genre>drama</genre>',
  '</movie>',
  '</movies>')
```

#### Explanation:

- The **xml2** library is loaded to handle XML data in R.
- The **stringr** library is loaded for string manipulation, though it's not used in this snippet.
- An XML string representing a list of movies is defined, including details like **title**, **director**, **year**, and **genre**.

### Section 2: Parsing the XML Document

```
doc <- read_xml(paste(xml_string, collapse = ''))
doc

## {xml_document}
## <movies>
## [1] <movie mins="126" lang="eng">\n <title>Good Will Hunting</title>\n <dir ...
## [2] <movie mins="106" lang="spa">\n <title>Y tu mama tambien</title>\n <dir ...
```

#### Explanation:

- The `read_xml` function from the `xml2` package is used to parse the XML string into an XML document object.
- The `paste` function with `collapse = ''` is used to concatenate the XML string into a single string before parsing.
- The `parsed` XML document is stored in the variable `doc`.

### Section 3: Navigating the XML Document

```
tu_mama <- xml_child(doc, search = 2)
tu_mama

## {xml_node}
## <movie mins="106" lang="spa">
## [1] <title>Y tu mama tambien</title>
## [2] <director>\n <first_name>Alfonso</first_name>\n <last_name>Cuaron</last ...
## [3] <year>2001</year>
## [4] <genre>drama</genre>

xml_children(tu_mama)

## {xml_nodeset (4)}
## [1] <title>Y tu mama tambien</title>
## [2] <director>\n <first_name>Alfonso</first_name>\n <last_name>Cuaron</last ...
## [3] <year>2001</year>
## [4] <genre>drama</genre>
```

#### Explanation

- The `xml_children` function lists all child nodes of the XML document.
- The `xml_child` function is used to select a specific child node by its index, in this case, the second movie.

### Section 4: Extracting director Information

```
director <- xml_child(tu_mama, "director")
director

## {xml_node}
## <director>
## [1] <first_name>Alfonso</first_name>
## [2] <last_name>Cuaron</last_name>

xml_contents(director)

## {xml_nodeset (2)}
## [1] <first_name>Alfonso</first_name>
```

```
## [2] <last_name>Cuaron</last_name>
```

```
xml_text(director)
```

```
## [1] "AlfonsoCuaron"
```

### Explanation

- The `xml_child` function is used again to select the “director” child node of the selected movie.
- The `xml_contents` function lists all nodes within the “director” node.
- The `xml_text` function extracts the text content of the “director” node, providing the **director’s name**.