The Foundations: Assignment and Data Types

In the Academy, we do not simply observe data; we command it. Before you can analyze complex signals, you must master the fundamental act of saving information into your research environment. This page covers the mechanics of the assignment operator and the primary data formats used in Base R. Include all code in your R script, `Chap1Lab.r`.

1. The Assignment Operator

In R, we use the left arrow operator (`<-`) to store data. It takes whatever is on the right side and places it into the variable name on the left.

Write the exact Base R code to assign your favorite number to a variable named `MyFavNum`.

2. The Character Trap

Read through Examples 1.8 and 1.21 in the text regarding the `BloodType` variable.

If a student types `BloodType <- AB` without quotation marks, explain exactly why R throws an error instead of saving the letters AB.

Bill the Statypus says: "Assignment binds symbols. Unquoted literals trigger failed lookups."

Sally the Statypus says: "Bill is absolutely exhausted from pulling an all-nighter on that new Statypus code, so please forgive him for being extra blunt today! He's just worried you'll forget your quotation marks for text. If you don't use them, R goes looking for a variable you haven't made yet."

3. Numeric vs. Character Identification

R classifies numbers as `num` and text strings as `chr`. Identify the type for each assignment below:

1. `MyAge <- 20` Type: _____
2. `MyGrade <- "A"` Type: _____
3. `PetName <- "Statypus"` Type: _____

Bill the Statypus says: "Try looking at your environment. Also, for you, `num` and `int` will always be the same thing."

The 1D Vector: Indexing and Extraction

A single data point is rarely enough for research. We typically work with **vectors**, which are 1-dimensional lists of data. To access specific parts of these lists, we use the bracket operator.

Statypus Insight: The Language of Symbols

In R, punctuation is as precise as math. Think of parentheses `( )` as **verbs**—they represent actions or functions like `sqrt()` or `log()`. Think of brackets `[ ]` as **nouns**—they represent addresses or locations within a data structure. If you try to use a bracket on a function, you're treating an action like a piece of furniture!

1. Definition 1.4

Navigate to Section 1.3.2. Record the exact definition of the **bracket operator** below:

2. Indexing Mechanics

Look at **Example 1.12** using the `Snickers` vector.

If you run `Snickers[15]`, R returns `50.8`. Explain what the `15` is telling R to do.

Write the syntax to extract the 42nd entry of the `Snickers` vector.

3. Identifying Intent

A student wants the 5th entry of a vector named `Heights`. They type `Heights(5)`. Based on **Example 1.23**, explain why R throws an error and what R *thinks* the student is trying to do.

Bill the Statypus says: "Functions utilize parentheses. Data structures require brackets. `sqrt[16]` is invalid."

Sally the Statypus says: "He's such a stickler for the rules! But he's trying to save you from a headache later. Just remember: use `( )` when you want R to *do* something, and `[ ]` when you want R to *find* something in a list."

The Logical Engine: Conditional Subsetting

Research often requires filtering data based on criteria rather than position. We do this by placing logical tests—questions with TRUE or FALSE answers—inside our brackets.

1. Logical Primitives

Type and run the following in an RStudio script:

```
> ToyData <- c( 10, 20, 30, 40 )
> LogVec <- c( TRUE, FALSE, TRUE, FALSE )
> ToyData[ LogVec ]
```

Explain why the code `ToyData[ LogVec ]` only returned 10 and 30. What did R do with the values aligned with `F`?

Bill the Statypus says: "Reassigning `T` or `F` corrupts evaluation."

Sally the Statypus says: "Bill is being protective of the language! Since `T` and `F` are shortcuts for TRUE and FALSE, if you ever name a variable `T`, R loses its "compass" for logic. Don't do it! This is why the old guard always use `TRUE` and `FALSE`."

2. The Hidden Map

Reference **Example 1.12**. When we run `Snickers[Snickers < 51]`, R creates a "hidden" vector of logical results.

Write the code to find Snickers bars weighing *greater than or equal to* 53g.

Write out the first 5 elements of the logical vector (`T/F`) R creates for the code you wrote above.

The Data Frame: Column Extraction

Most research data is stored in **data frames**—tables where each column is a vector. To analyze a specific variable, we must isolate it using the dollar sign operator.

Action Item: Loading Data

Run the `load(url(...))` command from Section 1.3.3. Check your Environment tab to ensure the datasets are loaded.

Bill the Statypus says: "`$` extracts atomic components."

Sally the Statypus says: "Exactly. It rips a single column out so we can work with it. Bill's been working so hard that he's forgotten how to say "please," but he's a genius with these data structures!"

1. The Assembly Line Anatomy

Diagram the following extraction code using arrows and labels:

```
BabyData1$weight[ BabyData1$Sex == "F" ]
```

☐ Label the part that isolates the column we want to *measure*.

☐ Label the part that creates the logical filter.

☐ Label the bridge that applies the filter to the data.

2. Using the Loaded Data

Look at the `BabyData1` you just loaded. Now, we want to find the length of gestation for all babies born to mothers who were 25 years old.

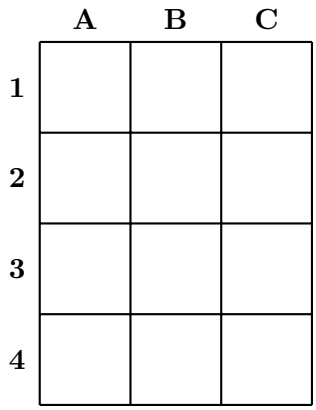
Write the code to extract the `gestation` column from `BabyData1` based on the `mom_age` column being 25. Include this code in your R script, `Chap1Lab.r`.

The 2D Restrictor: Rows and Columns

While the dollar sign extracts a single column, the bracket operator can slice a data frame in both dimensions. Because a data frame is a grid, we provide two instructions separated by a comma: `[row, column]`.

1. The 2D Grid

Below is a grid representing data frame `df` with 4 rows and 3 columns.



☐ Draw a solid box around the extraction of `df[ 2, ]`.

☐ Draw a dotted circle around the extraction of `df[ , 3 ]`.

☐ Put an 'X' in the cell extracted by `df[ 2, 3 ]`.

Bill the Statypus says: "Indices are `[row, column]`. Empty means unconstrained."

Sally the Statypus says: "He means if you leave the space after the comma blank, R gives you all the columns! It's like saying "I want this row, and I don't care which column you give me, so give me all of them." "

2. Real Data Extraction

Using the `BabyData1` in your Environment, write the code to extract the 10th row of the dataset.

Reflection: Command Line Syntax

How does this type of interface feel compared to spreadsheet programs you are used to?