

## Load necessary libraries

```
library(polite) library(rvest) library(dplyr) library(purrr) library(ggplot2)
```

## Function to scrape products from a given category URL

```
scrape_amazon_category <- function(category_url, category_name) { # Create a polite session session <-  
  bow(category_url, user_agent = "Educational")  
  
  # Scrape the page page <- scrape(session)  
  
  # Extract product details products <- page %>% html_nodes(".s-main-slot .s-result-item") %>% map_df(~  
  { tibble( Title = .x %>% html_node("h2") %>% html_text(trim = TRUE), Price = .x %>% html_node(".a-  
    price .a-offscreen") %>% html_text(trim = TRUE), Description = .x %>% html_node(".a-text-normal")  
    %>% html_text(trim = TRUE), Rating = .x %>% html_node(".a-icon-alt") %>% html_text(trim =  
    TRUE), Reviews = .x %>% html_node(".a-size-small .a-link-normal") %>% html_text(trim = TRUE),  
    Category = category_name # Add category name here ) })  
  
  return(products) }
```

## Example category URLs (you need to adjust these)

```
categories <- list( fishing = 'https://www.amazon.com/s?k=fishing', electronics = 'https://www.  
amazon.com/s?k=electronics', books = 'https://www.amazon.com/s?k=books', home_kitchen =  
'https://www.amazon.com/s?k=home+kitchen', clothing = 'https://www.amazon.com/s?k=clothing' )
```

## Initialize an empty data frame to store all products

```
all_products <- tibble()
```

## Loop through categories and scrape 30 products from each

```
for (category_name in names(categories)) { category_url <- categories[[category_name]] category_products  
<- scrape_amazon_category(category_url, category_name) all_products <- bind_rows(all_products, cate-  
gory_products)  
  
# Limit to 30 products if (nrow(all_products) > 30) { all_products <- all_products %>% slice(1:30) } }
```

## Convert Price and Rating to numeric for analysis

```
all_products$Price <- as.numeric(gsub(""," ", gsub(",", "", all_products$Price))) all_products$Rating <- as.numeric(gsub(" out of 5 stars", "",  
all_products$Rating))
```

## Display the extracted data

```
print(all_products)
```

## Plot Price and Ratings by Category

```
ggplot(all_products, aes(x = Title, y = Price, fill = Category)) + geom_bar(stat = "identity") + coord_flip()  
+ theme_minimal() + labs(title = "Price of Products by Category", x = "Product", y = "Price ($)")
```

```
ggplot(all_products, aes(x = Title, y = Rating, fill = Category)) + geom_bar(stat = "identity") + coord_flip()
+ theme_minimal() + labs(title = "Ratings of Products by Category", x = "Product", y = "Rating (out of
5)")
```

## Rank products by Price

```
ranked_by_price <- all_products %>% group_by(Category) %>% arrange(Price) %>% mutate(Price_Rank
= row_number())
```

## Rank products by Ratings

```
ranked_by_rating <- all_products %>% group_by(Category) %>% arrange(desc(Rating)) %>% mu-
tate(Rating_Rank = row_number())
```

## Display the ranked data

```
print(ranked_by_price) print(ranked_by_rating)
```