Tidy Tuesday

Week 36

AUTHOR
Cristian T

PUBLISHED
September 14, 2024

This week we are exploring the Stack Overflow Annual Developer Survey 2024!

This week's dataset is derived from the 2024 Stack Overflow Annual Developer Survey. Conducted in May 2024, the survey gathered responses from over 65,000 developers across seven key sections:

- Basic information
- Education, work, and career
- Tech and tech culture
- Stack Overflow community
- Artificial Intelligence (AI)
- Professional Developer Series - Not part of the main survey
- Thoughts on Survey

The dataset provided for this analysis focuses exclusively on the single-response questions from the main survey sections. Each categorical response in the survey has been integer-coded, with corresponding labels available in the crosswalk file.

`qname_levels_single_response_crosswalk.csv`

variable	class	description
qname	character	Categorical Question/Column Name in main data
level	integer	Integer index associated with each column response
label	character	Label associated with integer index

`stackoverflow_survey_questions.csv`

variable	class	description
qname	character	Categorical Question/Column Name in main data
question	character	Text of the question as it was presented to the respondent
variable	class	description

`stackoverflow_survey_single_response.csv`

variable	class	description
response_id	double	Respondent ID
main_branch	integer	Professional coding level of the respondent

variable	class	description
age	integer	Age
remote_work	integer	Current work situation
ed_level	integer	Highest education level completed
years_code	integer	Years the respondent has coded in total; More than 50 years coded as 51
years_code_pro	integer	Years the respondent has coded professionally; More than 50 years coded as 51
dev_type	integer	Best current-job description
org_size	integer	People in the organization
purchase_influence	integer	Level of influence in purchasing new technology at their organization
buildvs_buy	integer	How much customization was needed in most recent tool recommendation
country	character	Country in which the respondent lives
currency	character	Currency of the country
comp_total	double	Total compensation
so_visit_freq	integer	Stack Overflow visiting frequency
so_account	integer	Stack Overflow account status
so_part_freq	integer	Stack Overflow participation frequency
so_comm	integer	Whether the respondent considers themself a member of the Stack Overflow community?
ai_select	integer	Use AI in development process
ai_sent	integer	Stance on using AI tools as part of development workflow
ai_acc	integer	Trust in accuracy of AI as part of development workflow
ai_complex	integer	How well the respondent believes the AI tools they use in development workflows handle complex tasks
ai_threat	integer	Belief that AI is a threat to current job
survey_length	integer	Feeling about the length of the Stack Overflow Developer Survey this year

variable	class	description
survey_ease	integer	Ease of completion of this survey
converted_comp_yearly	double	Converted compensation
r_used	integer	Flag if respondent used R in the previous year
r_want_to_use	integer	Flag if respondent want to use R in the next year

```
# Load the tidytuesday package
suppressMessages(library(tidytuesdayR)) # For accessing TidyTuesday datasets
suppressMessages(library(skimr)) # For summary and descriptive statistics
suppressMessages(library(tidyverse)) # For data manipulation and visualization
suppressMessages(library(dplyr)) # For data manipulation and transformation
suppressMessages(library(ggplot2)) # For data visualization
suppressMessages(library(RColorBrewer)) # For color palettes in visualizations
suppressMessages(library(mdsr)) # For spatial analysis
suppressMessages(library(tidytext)) # For text mining and analysis
suppressMessages(library(textstem)) # For lemmatization


# Load the current week's dataset
tuesdata <- tidytuesdayR::tt_load('2024-09-03')
```

```
Downloading file 1 of 3: `qname_levels_single_response_crosswalk.csv`
Downloading file 2 of 3: `stackoverflow_survey_questions.csv`
Downloading file 3 of 3: `stackoverflow_survey_single_response.csv`
```

```
# Extract datasets from the TidyTuesday dataset
qname_levels_single_response_crosswalk <- tuesdata$qname_levels_single_response_cross
stackoverflow_survey_questions <- tuesdata$stackoverflow_survey_questions
stackoverflow_survey_single_response <- tuesdata$stackoverflow_survey_single_response

# Rename datasets
survey_questions <- stackoverflow_survey_questions
survey_responses <- stackoverflow_survey_single_response
qname_levels <- qname_levels_single_response_crosswalk

# Explore the structure of the dataset
str(survey_questions) # Display the structure of 'survey_questions' (type of variable
```

```
spc_tbl_ [24 × 2] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ qname   : chr [1:24] "main_branch" "age" "remote_work" "ed_level" ...
 $ question: chr [1:24] "Which of the following options best describes you today? For the
purpose of this survey, a developer is \"someo"| __truncated__ "What is your age?*" "Which
best describes your current work situation?" "Which of the following best describes the
highest level of formal education that you've completed? *" ...
 - attr(*, "spec")=
  .. cols(
  ..   qname = col_character(),
  ..   question = col_character()
```

```
  .. )
 - attr(*, "problems")=<externalptr>
```

```r
#skim(survey_questions)

# Explore the structure of the survey responses dataset
str(survey_responses) # Display the structure of 'survey_responses'
```

```
spc_tbl_ [65,437 × 28] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ response_id          : num [1:65437] 1 2 3 4 5 6 7 8 9 10 ...
 $ main_branch          : num [1:65437] 1 1 1 2 1 4 3 2 4 1 ...
 $ age                  : num [1:65437] 8 3 4 1 1 8 3 1 4 3 ...
 $ remote_work          : num [1:65437] 3 3 3 NA NA NA 3 NA 2 3 ...
 $ ed_level             : num [1:65437] 4 2 3 7 6 4 5 6 5 3 ...
 $ years_code           : num [1:65437] NA 20 37 4 9 10 7 1 20 15 ...
 $ years_code_pro       : num [1:65437] NA 17 27 NA NA NA 7 NA NA 11 ...
 $ dev_type             : num [1:65437] NA 16 10 16 16 33 1 33 1 16 ...
 $ org_size             : num [1:65437] NA NA NA NA NA NA NA NA NA NA ...
 $ purchase_influence   : num [1:65437] NA NA NA NA NA NA NA NA NA NA ...
 $ buildvs_buy          : num [1:65437] NA NA NA NA NA NA NA NA NA NA ...
 $ country              : chr [1:65437] "United States of America" "United Kingdom of Great
Britain and Northern Ireland" "United Kingdom of Great Britain and Northern Ireland" "Canada"
...
 $ currency             : chr [1:65437] NA NA NA NA ...
 $ comp_total           : num [1:65437] NA NA NA NA NA NA NA NA NA NA ...
 $ so_visit_freq        : num [1:65437] NA 5 5 3 5 5 3 4 5 2 ...
 $ so_account           : num [1:65437] NA 3 3 1 3 3 3 1 3 3 ...
 $ so_part_freq         : num [1:65437] NA 6 6 NA 6 6 3 NA 6 5 ...
 $ so_comm              : num [1:65437] NA 5 5 3 5 5 5 2 5 6 ...
 $ ai_select            : num [1:65437] 3 1 1 3 1 3 1 3 1 3 ...
 $ ai_sent              : num [1:65437] 5 NA NA 5 NA 1 NA 2 NA 2 ...
 $ ai_acc               : num [1:65437] NA NA NA 5 NA 5 NA 4 NA 3 ...
 $ ai_complex           : num [1:65437] NA NA NA 1 NA 2 NA 1 NA 1 ...
 $ ai_threat            : num [1:65437] NA NA NA 2 NA 2 NA 3 NA 1 ...
 $ survey_length        : num [1:65437] NA NA 1 2 3 1 2 1 1 2 ...
 $ survey_ease          : num [1:65437] NA NA 2 2 2 2 3 1 3 2 ...
 $ converted_comp_yearly: num [1:65437] NA NA NA NA NA NA NA NA NA NA ...
 $ r_used               : num [1:65437] NA 0 0 0 0 0 1 0 0 0 ...
 $ r_want_to_use        : num [1:65437] NA 0 0 0 0 0 1 0 0 0 ...
 - attr(*, "spec")=
  .. cols(
  ..   response_id = col_double(),
  ..   main_branch = col_double(),
  ..   age = col_double(),
  ..   remote_work = col_double(),
  ..   ed_level = col_double(),
  ..   years_code = col_double(),
  ..   years_code_pro = col_double(),
  ..   dev_type = col_double(),
  ..   org_size = col_double(),
  ..   purchase_influence = col_double(),
  ..   buildvs_buy = col_double(),
  ..   country = col_character(),
  ..   currency = col_character(),
```

```
  ..   comp_total = col_double(),
  ..   so_visit_freq = col_double(),
  ..   so_account = col_double(),
  ..   so_part_freq = col_double(),
  ..   so_comm = col_double(),
  ..   ai_select = col_double(),
  ..   ai_sent = col_double(),
  ..   ai_acc = col_double(),
  ..   ai_complex = col_double(),
  ..   ai_threat = col_double(),
  ..   survey_length = col_double(),
  ..   survey_ease = col_double(),
  ..   converted_comp_yearly = col_double(),
  ..   r_used = col_double(),
  ..   r_want_to_use = col_double()
  .. )
 - attr(*, "problems")=<externalptr>
```

```
skim(survey_responses) # Provide detailed summary statistics for 'survey_responses' (
```

Variable type: numeric

var	n	na	mean	sd	p0	p25	p50	p75	p100
response_id	65437	0	3.271900e+04	1.889018e+04	1	16360	32719	49078.0	6.54370e+04
main_branch	65437	0	1.500000e+00	1.020000e+00	1	1	1	1.0	5.00000e+00
age	65437	0	2.630000e+00	1.580000e+00	1	2	2	3.0	8.00000e+00
remote_work	65437	10631	1.960000e+00	8.900000e-01	1	1	2	3.0	3.00000e+00
ed_level	65437	4653	3.510000e+00	1.930000e+00	1	2	3	5.0	8.00000e+00
years_code	65437	5568	1.420000e+01	1.066000e+01	0	6	11	20.0	5.10000e+01
years_code_pro	65437	13827	1.018000e+01	9.110000e+00	0	3	7	15.0	5.10000e+01
dev_type	65437	5992	1.717000e+01	7.750000e+00	1	12	16	19.0	3.40000e+01
org_size	65437	17957	4.770000e+00	2.480000e+00	1	3	5	6.0	1.00000e+01
purchase_influence	65437	18031	2.190000e+00	7.700000e-01	1	2	2	3.0	3.00000e+00
buildvs_buy	65437	22079	1.600000e+00	8.000000e-01	1	1	1	2.0	3.00000e+00
comp_total	65437	31697	2.963841e+145	5.444117e+147	0	60000	110000	250000.0	1.00000e+150
so_visit_freq	65437	5901	2.510000e+00	1.270000e+00	1	2	2	3.0	5.00000e+00
so_account	65437	5877	2.600000e+00	7.500000e-01	1	3	3	3.0	3.00000e+00
so_part_freq	65437	20200	4.020000e+00	1.430000e+00	1	4	5	5.0	6.00000e+00
so_comm	65437	6274	3.360000e+00	1.840000e+00	1	2	3	5.0	6.00000e+00
ai_select	65437	4530	2.370000e+00	8.500000e-01	1	2	3	3.0	3.00000e+00
ai_sent	65437	19564	2.390000e+00	1.680000e+00	1	1	2	4.0	6.00000e+00
ai_acc	65437	28135	3.850000e+00	1.210000e+00	1	3	4	5.0	5.00000e+00
ai_complex	65437	28416	2.230000e+00	1.110000e+00	1	1	2	3.0	5.00000e+00
ai_threat	65437	20748	1.920000e+00	5.600000e-01	1	2	2	2.0	3.00000e+00

var	n	na	mean	sd	p0	p25	p50	p75	p100
survey_length	65437	9255	1.330000e+00	5.000000e-01	1	1	1	2.0	3.00000e+00
survey_ease	65437	9199	2.410000e+00	5.500000e-01	1	2	2	3.0	3.00000e+00
converted_comp_yearly	65437	42002	8.615529e+04	1.867570e+05	1	32712	65000	107971.5	1.62566e+07
r_used	65437	5692	4.000000e-02	2.000000e-01	0	0	0	0.0	1.00000e+00
r_want_to_use	65437	9685	4.000000e-02	1.900000e-01	0	0	0	0.0	1.00000e+00

```
# View the first few rows of the dataset
head(survey_questions)
```

qname
<chr>
main_branch
age
remote_work
ed_level
years_code
years_code_pro

6 rows | 1-1 of 2 columns

```
head(survey_responses)
```

response_id	main_branch	age	remote_work	ed_level	years_code
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	1	8	3	4	*NA*
2	1	3	3	2	20
3	1	4	3	3	37
4	2	1	*NA*	7	4
5	1	1	*NA*	6	9
6	4	8	*NA*	4	10

6 rows | 1-6 of 28 columns

```
# Export data
#write.csv(qname_levels_single_response_crosswalk, "qname_levels_single_response_cros
#write.csv(stackoverflow_survey_questions, "stackoverflow_survey_questions.csv", row.
#write.csv(stackoverflow_survey_single_response, "stackoverflow_survey_single_respons


#tidytuesdayR::use_tidytemplate()
```

Clean the data

```
# Skim showed a lot of missing values so we will remove rows with missing info
```

```
#survey_responses_clean <- survey_responses %>% drop_na()
```

Merge the data

We will merge the `qname_levels` and `survey_questions` data frames to get a full picture of the questions.

```
# Merge qname_levels and survey_questions on the 'qname' column
merged_questions <- merge(qname_levels, survey_questions, by="qname", all=TRUE) %>%
  select(level, question, qname, label, everything())

# Inspect the merged data
head(merged_questions)
```

	level	question	qname	label
	<dbl>	<chr>	<chr>	<chr>
1	1	What is your age?*	age	18-24 years old
2	2	What is your age?*	age	25-34 years old
3	3	What is your age?*	age	35-44 years old
4	4	What is your age?*	age	45-54 years old
5	5	What is your age?*	age	55-64 years old
6	6	What is your age?*	age	65 years or older

6 rows

```
# Remove HTML tags from the 'question' column using regular expressions
merged_questions$question <- gsub("<[^>]+>", "", merged_questions$question)

# Add missing level and label for rows 34-36
merged_questions[34, "level"] <- 1
merged_questions[34, "label"] <- "income"
merged_questions[35, "level"] <- 2
merged_questions[35, "label"] <- "residence"
merged_questions[36, "level"] <- 3
merged_questions[36, "label"] <- "day to day currency"

# Add missing level and label for rows 126 and 127
merged_questions[126, "level"] <- 1
merged_questions[126, "label"] <- "total experience"
merged_questions[127, "level"] <- 2
merged_questions[127, "label"] <- "professional experience"

# Inspect the modified data
head(merged_questions[c(34:36, 126:127), ])
```

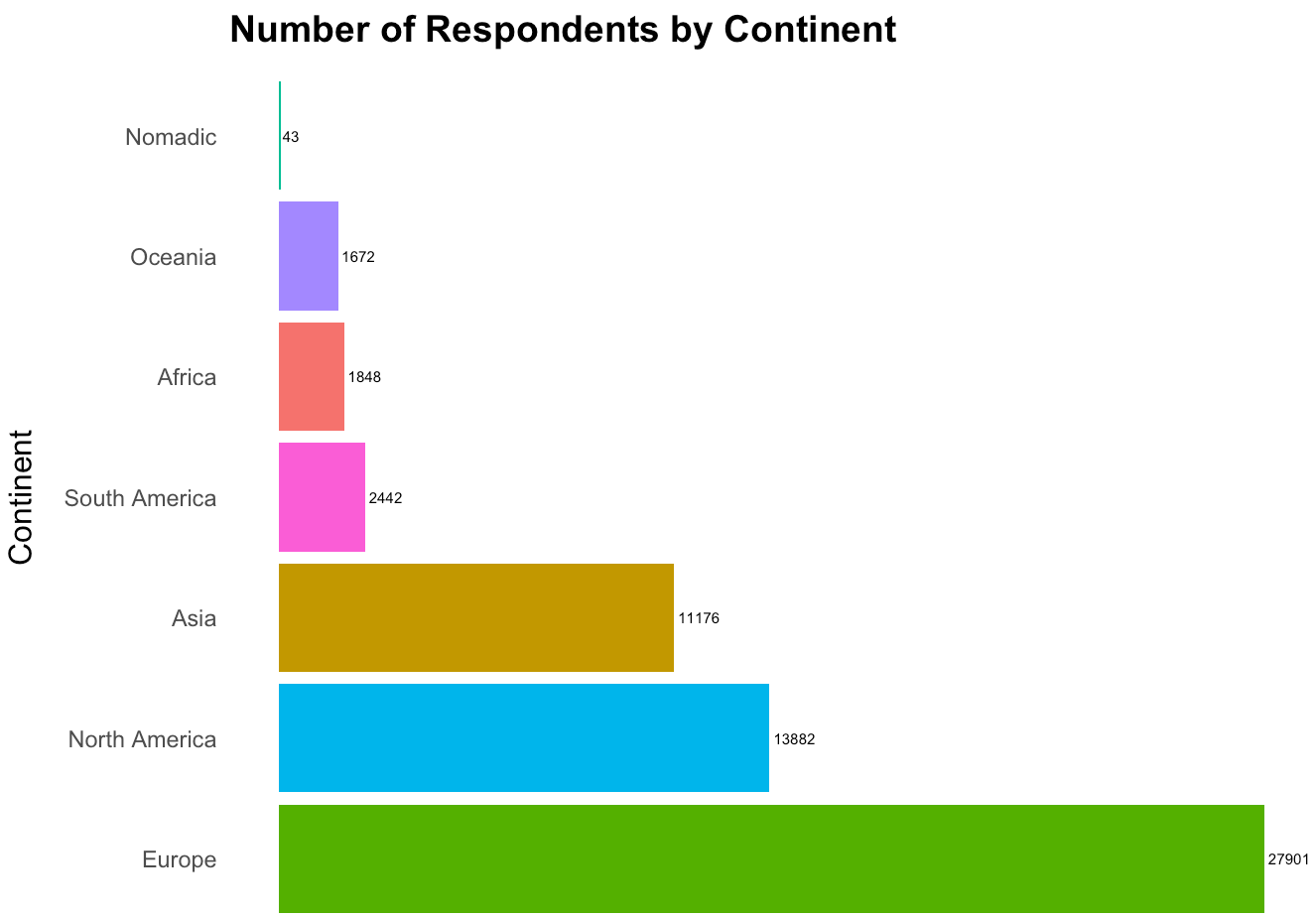
	level
	<dbl>
34	1
35	2
36	3
126	1
127	2

5 rows | 1-2 of 5 columns

EDA

```r
# Load a CSV file that maps countries to continents
manual_continent_mapping <- read.csv("/Users/cristianthirteen/Downloads/Notre Dame/SP

# Rename the "Country" column in the manual_continent_mapping to "country" for consis
manual_continent_mapping <- manual_continent_mapping %>%
  rename(country = Country)

# Merge manual_continent_mapping with survey_responses by the "country" column
survey_responses <- merge(manual_continent_mapping, survey_responses, by = "country",


# Create a named vector for the country corrections
country_corrections <- c(
  "Brunei Darussalam" = "Brunei",
  "Burkina Faso" = "Burkina",
  "Congo, Democratic Republic of" = "Congo",
  "Congo, Republic of the..." = "Congo",
  "Democratic Republic of the Congo" = "Congo",
  "Democratic People's Republic of Korea" = "North Korea",
  "Iran, Islamic Republic of..." = "Iran",
  "Lao People's Democratic Republic" = "Laos",
  "Libyan Arab Jamahiriya" = "Libya",
  "Micronesia, Federated States of..." = "Micronesia",
  "Myanmar" = "Burma (Myanmar)",
  "Nomadic" = "Nomadic",
  "Republic of Korea" = "South Korea",
  "Republic of Moldova" = "Moldova",
  "Republic of North Macedonia" = "North Macedonia",
  "Syrian Arab Republic" = "Syria",
  "United Kingdom of Great Britain and Northern Ireland" = "United Kingdom",
  "United Republic of Tanzania" = "Tanzania",
  "Venezuela, Bolivarian Republic of..." = "Venezuela",
  "Viet Nam" = "Vietnam"
)

# Apply the country corrections to the survey_responses dataset
survey_responses<- survey_responses %>%
  mutate(country = recode(country, !!!country_corrections))

# Apply corrections for missing continent names based on the country values
survey_responses <- survey_responses %>%
     mutate(Continent = case_when(
       is.na(Continent) & country == "Brunei" ~ "Asia",
       is.na(Continent) & country == "Burkina" ~ "Africa",
       is.na(Continent) & country == "Congo" ~ "Africa",
       is.na(Continent) & country == "North Korea" ~ "Asia",
       is.na(Continent) & country == "Iran" ~ "Asia",
       is.na(Continent) & country == "Laos" ~ "Asia",
       is.na(Continent) & country == "Libya" ~ "Africa",
```

```r
      is.na(Continent) & country == "Micronesia" ~ "Oceania",
      is.na(Continent) & country == "Burma (Myanmar)" ~ "Asia",
      is.na(Continent) & country == "Nomadic" ~ "Nomadic",
      is.na(Continent) & country == "South Korea" ~ "Asia",
      is.na(Continent) & country == "Moldova" ~ "Europe",
      is.na(Continent) & country == "North Macedonia" ~ "Europe",
      is.na(Continent) & country == "Syria" ~ "Asia",
      is.na(Continent) & country == "United Kingdom" ~ "Europe",
      is.na(Continent) & country == "Tanzania" ~ "Africa",
      is.na(Continent) & country == "Venezuela" ~ "South America",
      is.na(Continent) & country == "Vietnam" ~ "Asia",
      TRUE ~ Continent
    ))

# Group survey_responses by Continent and count the number of respondents per contine
continent_counts <- survey_responses %>%
  filter(!is.na(Continent)) %>%
  group_by(Continent) %>%
  summarise(count = n()) %>%
  arrange(desc(count)) %>%
  mutate(percentage = (count / sum(count)) * 100)

# Filter rows where country is "Congo"
# survey_responses %>% filter(country == "Congo")


# Plot number of respondents by country
ggplot(continent_counts, aes(x = reorder(Continent, -count), y = count, fill = Contir
  geom_bar(stat = "identity", show.legend = FALSE) +
  geom_text(aes(label = count), hjust = -0.1, size = 2) +
  labs(title = "Number of Respondents by Continent",
       x = "Continent",
       y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.title.x = element_blank(),
        axis.title.y = element_text(size = 12, margin = margin(r = 10)),
        plot.title = element_text(size = 14, face = "bold"),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank()) +
  coord_flip()
```



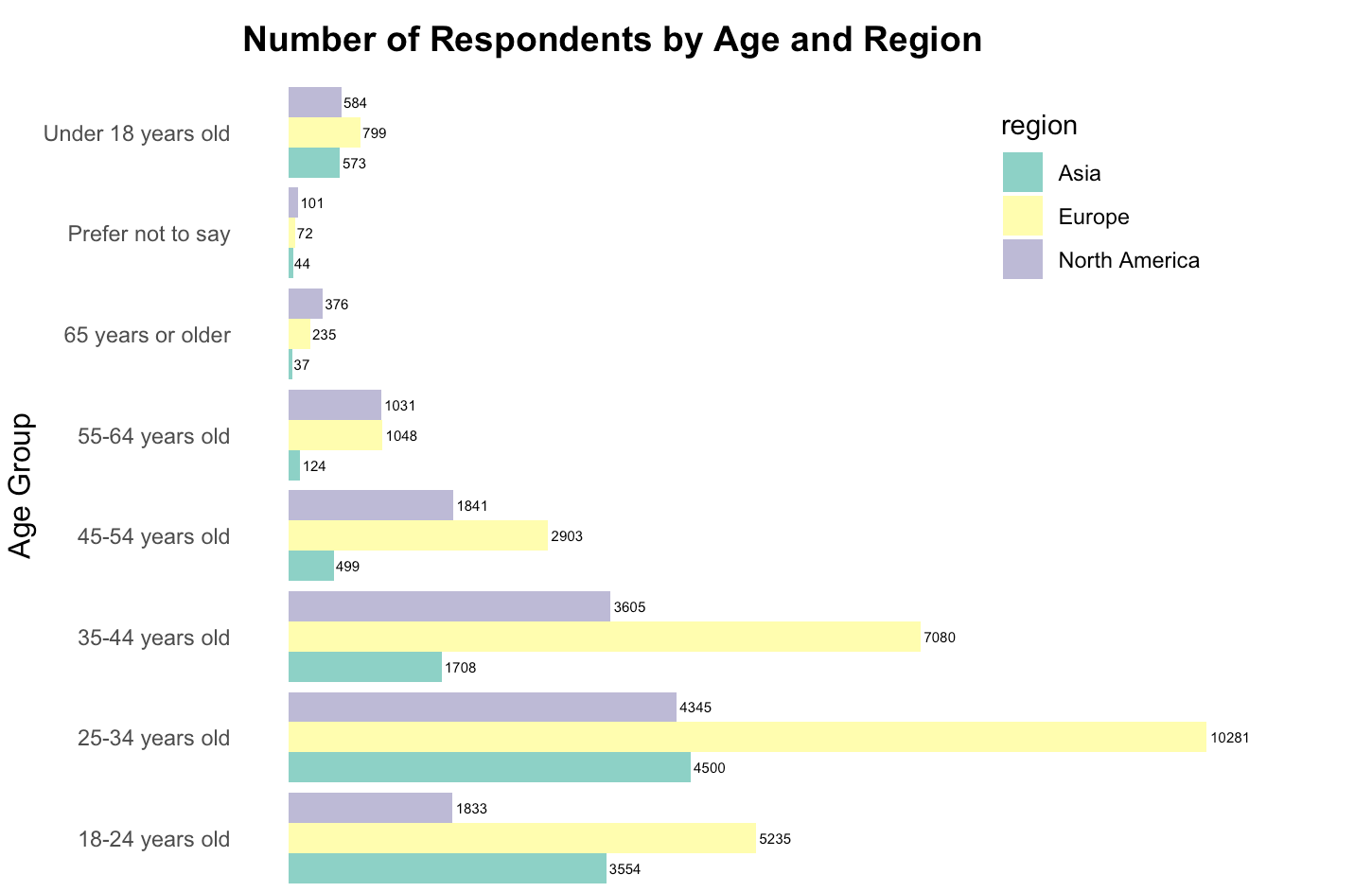

```
#write.csv(survey_responses, "survey_responses.csv", row.names = FALSE)
```

The survey data reveals that nearly half (47.3%) of the respondents are from Europe, making it the largest represented continent. North America follows with 23.5% of respondents, and Asia accounts for 19.0%. South America, Africa, and Oceania have smaller representations, with 4.14%, 3.13%, and 2.84% of respondents, respectively. A small fraction (0.073%) of respondents identified as “Nomadic,” reflecting a respondent population that does not associate with a specific continent.

Plot 1 Top 3 Counties by Age

```
# Rename age groups
age_groups <- survey_responses %>%
  mutate(age = factor(age, levels = 1:8, labels = c(
    "18-24 years old",
    "25-34 years old",
    "35-44 years old",
    "45-54 years old",
    "55-64 years old",
    "65 years or older",
    "Prefer not to say",
    "Under 18 years old"
  )))

# Filter and count by continent and age
continent_age_counts <- age_groups %>%
  filter(!is.na(dev_type)) %>% # Exclude rows with missing 'dev_type'
  filter(Continent %in% c("North America", "Europe", "Asia")) %>%  # Select only the
```

```r
  mutate(region = case_when(
    Continent == "North America" ~ "North America",
    Continent == "Europe" ~ "Europe",
    Continent == "Asia" ~ "Asia"
  )) %>%
  group_by(region, age) %>% # Group by region and age
  summarise(count = n(), .groups = 'drop') %>% # Count the number of responses in eac
  mutate(percentage = (count / sum(count)) * 100)

# Create a bar plot showing the number of respondents by age and region
ggplot(continent_age_counts, aes(x = age, y = count, fill = region)) +
  geom_bar(stat = "identity", position = "dodge") +
  geom_text(aes(label = count), position = position_dodge(width = 0.9),hjust = -0.1,s
  scale_fill_manual(values = brewer.pal(10, "Set3")) +
  labs(title = "Number of Respondents by Age and Region",
       x = "Age Group",
       y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.title.x = element_blank(),
          axis.title.y = element_text(size = 12),
          plot.title = element_text(size = 14, face = "bold"),
          panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          legend.position = c(0.85, 0.85)) +
     coord_flip()
```

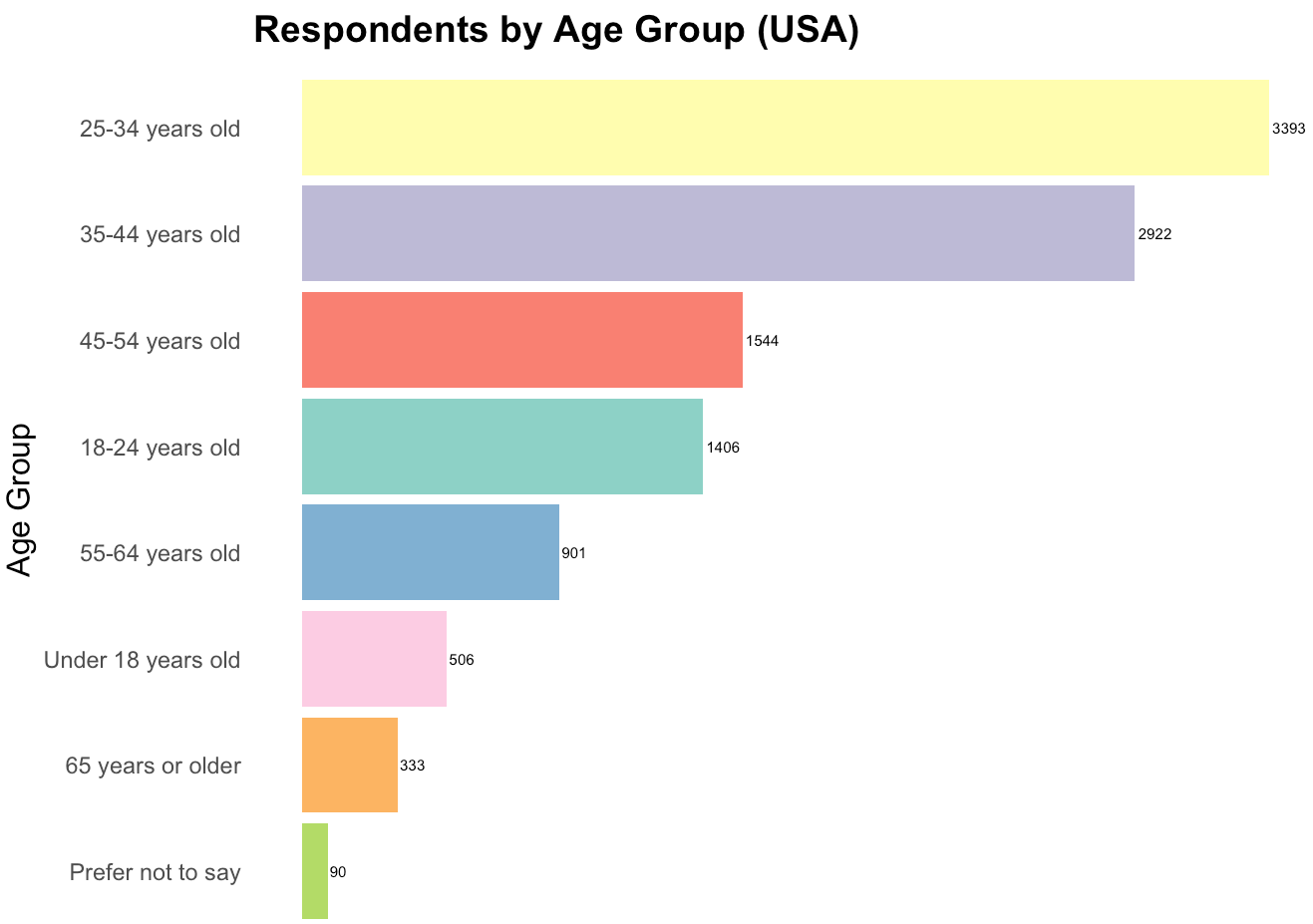


Across all regions, most respondents fall into the 25-34-year-old age group. Europe stands out with a notably higher percentage in this age bracket (19.60%) compared to Asia (8.59%) and North America (8.29%). North America has a relatively higher percentage of respondents in the 35-44-year-old group (6.88%) compared to both Asia and Europe. Additionally, there is a significant representation in the 18-24-year-old group across all regions, reflecting the common trend of younger individuals being more engaged with online surveys and tech-related topics.

In contrast, older age groups (55 years or older) show generally lower representation across all regions, which might indicate lower engagement or participation rates from older individuals. The percentage of respondents choosing “Prefer not to say” about their age is relatively small across all regions, suggesting that most respondents are comfortable disclosing their age.

Plot 2 Counts by Age Group (USA)

```
# Filter for USA and aggregate counts by age
usa_age_counts <- age_groups %>%
  filter(country == "United States of America") %>%
  group_by(age) %>%
  summarise(count = n(), .groups = 'drop') %>%
  mutate(percentage = (count / sum(count)) * 100)

# Create a bar plot
ggplot(usa_age_counts, aes(x = reorder(age, count), y = count, fill = age)) +
     geom_bar(stat = "identity") +
     scale_fill_manual(values = brewer.pal(10, "Set3")) +
  geom_text(aes(label = count), hjust = -0.1, size = 2) +
     labs(title = "Respondents by Age Group (USA)",
          x = "Age Group",
          y = "Count",
          fill = "Age Group") +
     theme_minimal() +
     theme(axis.text.x = element_blank(),
        axis.title.x = element_blank(),
           axis.title.y = element_text(size = 12),
           plot.title = element_text(size = 14, face = "bold"),
           panel.grid.major = element_blank(),
           panel.grid.minor = element_blank(),
           legend.position = "none") +  # Remove the legend
     coord_flip()
```



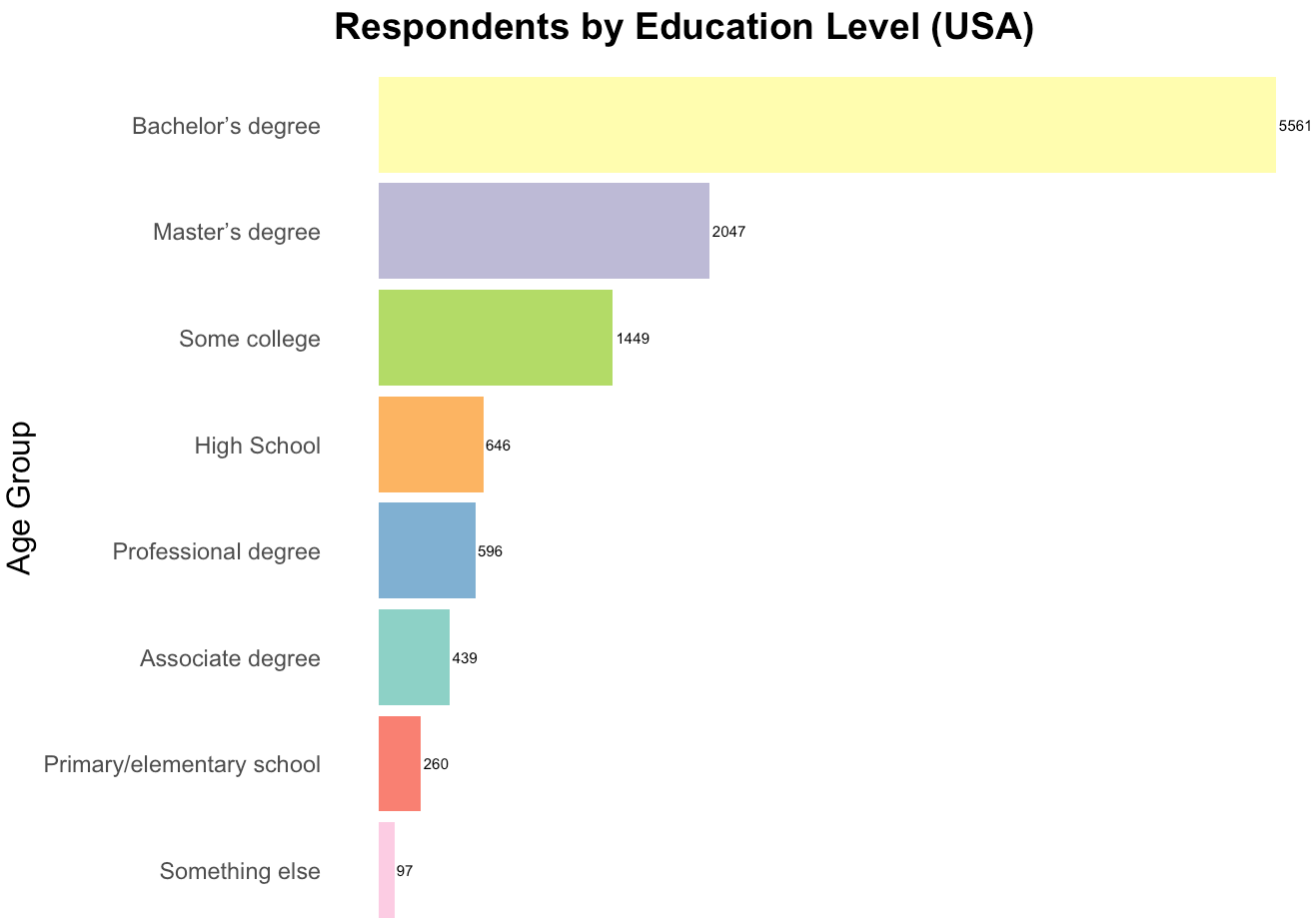
In the US, the age distribution of respondents reveals a significant concentration in the 25-34 age group, constituting 30.6% of the total respondents. The 35-44 age group follows closely, making up 26.3%. 18-24-year-olds account for 12.7%, showing notable but lesser participation than older groups. 45-54 years olds represent 13.9%, maintaining a substantial middle-aged presence. 55-64 years olds make up 8.12%, and those under 18 constitute 4.56%. The 65 and older group is the smallest at 3.00%. Additionally, 0.811% of respondents chose not to disclose their age.

Overall, 56.9% of respondents fall within the 25-44 age range, indicating that the survey predominantly reflects individuals in their prime working years.

Plot 3 Counts by Education Level (USA)

```
# Rename education
education_level <- survey_responses %>%
  mutate(ed_level = factor(ed_level, levels = 1:8, labels = c(
    "Associate degree",
    "Bachelor's degree",
    "Master's degree",
    "Primary/elementary school",
    "Professional degree",
    "High School",
    "Some college",
    "Something else"
  )))


# Filter for USA and aggregate counts by education
```

```r
usa_edu_counts <- education_level %>%
  filter(country == "United States of America") %>%
  group_by(ed_level) %>%
  summarise(count = n(), .groups = 'drop') %>%
  mutate(percentage = (count / sum(count)) * 100)

# Create a bar plot
ggplot(usa_edu_counts, aes(x = reorder(ed_level, count), y = count, fill = ed_level)) +
     geom_bar(stat = "identity") +
     scale_fill_manual(values = brewer.pal(10, "Set3")) +
  geom_text(aes(label = count), hjust = -0.1, size = 2) +
     labs(title = "Respondents by Education Level (USA)",
          x = "Age Group",
          y = "Count",
          fill = "Age Group") +
     theme_minimal() +
     theme(axis.text.x = element_blank(),
        axis.title.x = element_blank(),
           axis.title.y = element_text(size = 12),
           plot.title = element_text(size = 14, face = "bold"),
           panel.grid.major = element_blank(),
           panel.grid.minor = element_blank(),
           legend.position = "none") +  # Remove the legend
     coord_flip()
```



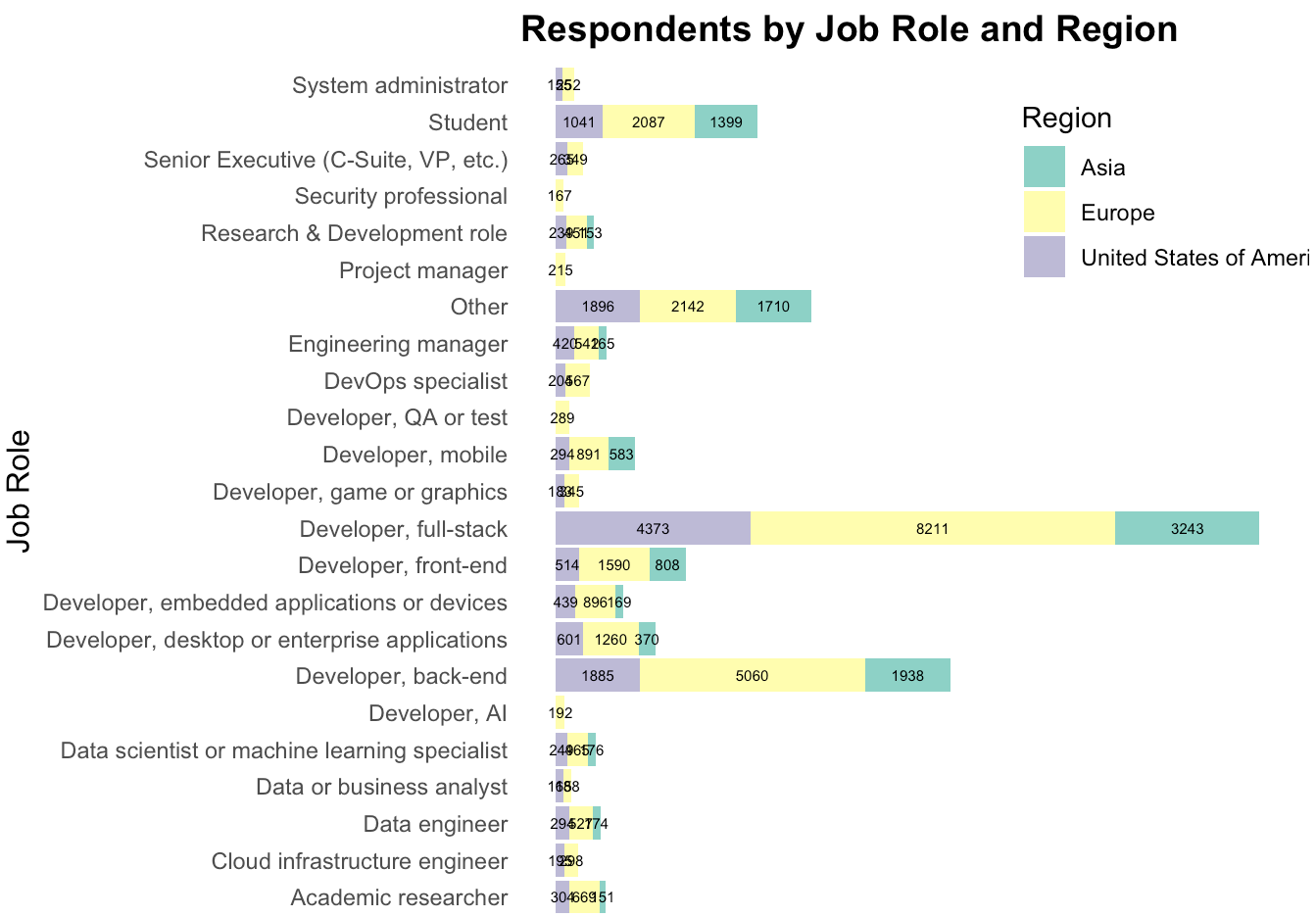
Our data for the US shows a predominance of Bachelor's Degree Holders, with 50.1% holding a Bachelor's degree. We also see a significant amount of graduate degrees, with Master's degree holders making up 18.4% of the respondents, suggesting that the survey sample is highly educated. We also see people with

associate degrees (3.96%), professional degrees (5.37%), high school diplomas (5.82%), and some college degrees (13.1%). The distribution shows a clear preference towards higher education among respondents.

Plot 4 Top 3 Countries by Job Role

```r
# Rename job roles
dev_roles <- survey_responses %>%
  mutate(dev_type = factor(dev_type, levels = 1:34, labels = c(
    "Academic researcher",
    "Blockchain",
    "Cloud infrastructure engineer",
    "Data engineer",
    "Data or business analyst",
    "Data scientist or machine learning specialist",
    "Database administrator",
    "Designer",
    "Developer Advocate",
    "Developer Experience",
    "Developer, AI",
    "Developer, back-end",
    "Developer, desktop or enterprise applications",
    "Developer, embedded applications or devices",
    "Developer, front-end",
    "Developer, full-stack",
    "Developer, game or graphics",
    "Developer, mobile",
    "Developer, QA or test",
    "DevOps specialist",
    "Educator",
    "Engineer, site reliability",
    "Engineering manager",
    "Hardware Engineer",
    "Marketing or sales professional",
    "Other",
    "Product manager",
    "Project manager",
    "Research & Development role",
    "Scientist",
    "Security professional",
    "Senior Executive (C-Suite, VP, etc.)",
    "Student",
    "System administrator"
  )))

# Filter for top 2 continents, including United States and count respondents by job r
continent_age_counts_dev <- dev_roles %>%
  filter(Continent %in% c("North America", "Europe", "Asia")) %>%
  mutate(region = case_when(
    Continent == "North America" ~ "United States of America",
    Continent == "Europe" ~ "Europe",
    Continent == "Asia" ~ "Asia"
  )) %>%
  group_by(region, dev_type) %>%
  summarise(count = n(), .groups = 'drop') %>%
```

```r
  mutate(percentage = (count / sum(count)) * 100)

# Group less common job roles into "Other"
threshold <- 150  # Threshold for grouping
grouped_data <- continent_age_counts_dev %>%
  filter(!is.na(dev_type)) %>%
  mutate(dev_type = if_else(count < threshold, "Other", as.character(dev_type))) %>%
  group_by(region, dev_type) %>%
  summarise(count = sum(count), .groups = 'drop') %>%
  mutate(percentage = (count / sum(count)) * 100)


# Create the plot
ggplot(grouped_data, aes(x = dev_type, y = count, fill = region)) +
  geom_bar(stat = "identity", position = "stack") +  # Change to "stack" for stacked
  geom_text(aes(label = count), position = position_stack(vjust = 0.5), size = 2) +
  scale_fill_manual(values = brewer.pal(10, "Set3")) +
  labs(title = "Respondents by Job Role and Region",
       x = "Job Role",
       y = "Count",
       fill = "Region") +
  theme_minimal() +
  theme(axis.text.x = element_blank(),
        axis.title.x = element_blank(),
        axis.title.y = element_text(size = 12),
        plot.title = element_text(size = 14, face = "bold"),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        legend.position = c(0.85, 0.85)) +
  coord_flip()  # Flip coordinates for better label readability
```



In analyzing job roles across different regions, it is evident that Europe has the highest proportion of full-stack developers (15.7%), reflecting a strong demand for professionals skilled in front-end and back-end development. Asia also shows a significant presence of full-stack developers (6.19%) and a notable percentage of back-end developers (3.70%), indicating a balanced focus on both roles, along with a substantial proportion of data scientists or machine learning specialists (3.36%). In the United States, full-stack developers (8.34%) and back-end developers (3.60%) are prominent, with a considerable share in the “Other” category (3.62%) that encompasses diverse roles. Across all regions, less common roles like academic researchers, blockchain developers, and cloud infrastructure engineers have low representation, generally below 1%. Students constitute a notable portion of respondents in Europe (3.98%) and the United States (1.99%), while research and development roles are significantly represented, particularly in Europe.