

# Regular Expressions

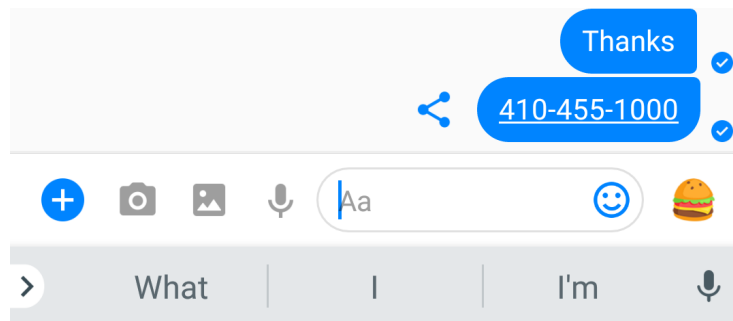
## Introduction

# Regular Expressions

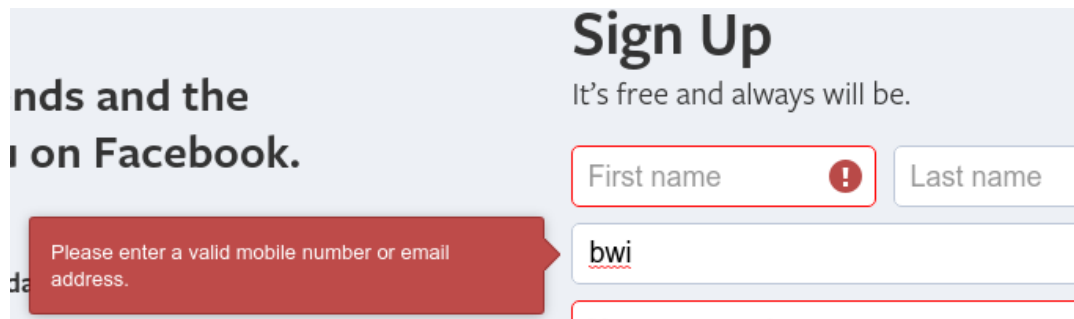
- Formally, they define a regular language
  - One that can be recognized with a Deterministic Finite Automata (DFA)
- Informally, they define a pattern to be matched
  - Later we will talk about more advanced uses, like substitution
- Regular expression is commonly shorted to **regexp**, **regex**, or **re**

# Applications of RegEx Matching

- Finding duplicated words in text
  - I went to **the the** store
- Recognizing dates and phone numbers in text



- Validating input

A screenshot of a Facebook sign-up form. The title is "Sign Up" and the subtitle is "It's free and always will be." There are two input fields for "First name" and "Last name". The "First name" field has a red border and a red exclamation mark icon, indicating an error. Below the "First name" field is a text input field containing "bwi". A red error message box is visible on the left side of the form, stating "Please enter a valid mobile number or email address." The form is set against a light gray background with some text partially visible on the left.

## Applications of RegEx Substitution

- Removing duplicate words
  - I went to **the the** store → I went to **the** store
- Fixing common case errors
  - We will be learning **javascript** this semester → We will be learning **JavaScript** this semester
- Reformatting dates and telephone numbers
  - 1-410-455-1000 → +1 (410) 455-1000
- Inserting links around phone numbers, emails, etc.

## RegEx's you already might use

- File searching (file globbing)
  - `ls *.png`
  - `rm *.bak`
  - `cp Lecture???.html ../`
- Find & Replace in word processors/text editors
  - Most allow you to select to use RegEx matching or not
- Search Engines
  - Either allow regex's or borrow concepts from them

## Why Regular Expressions *Seem* Intimidating

- Cryptic and compact
- Whitespace sensitive (typically)
- No standard
  - Differences between implementations
- Some characters are overloaded
- Multiple solutions usually exist
- Can be time consuming to iteratively tune a regex

# Programming Language Support

- Almost all programming languages support regex's, either natively or through a module/library

- C++

```
#include <regex>
```

- Python

```
import re
```

- Java

```
import java.util.regex.Pattern;
```

## Perl & RegExs

- Perl is a programming language developed with text processing in mind
  - Current version is Perl 5, which is available on GL
- Regular Expressions are a native part of Perl
- The syntax developed for Perl is extremely popular and many other languages support it
  - Kown as Perl Compatable Regular Expressions (PCRE)



## Perl in This Class

- We will use Perl to learn regular expressions, but you are only responsible for a very small subset of perl
- All code will be of the format

```
foreach my $n (@names) {  
    print $n if $n =~ /REGEX_HERE/;  
}
```

or

```
while (<>) {  
    print if /REGEX_HERE/;  
}
```

## Data for today's examples

- We will be using the current [Billboard Hot 100 Singles Chart](#)
- For convenience, I have stored each song and artist as a string in a large array

```
In [33]: say join "\n", @songs[0..5];
```

```
God's Plan by Drake  
Perfect by Ed Sheeran  
Havana by Camila Cabello Featuring Young Thug  
Rockstar by Post Malone Featuring 21 Savage  
Finesse by Bruno Mars & Cardi B  
Bad At Love by Halsey
```

```
Out[33]: 1
```

# Literals

- Literals match exactly
  - Every character matches only with itself

```
In [37]: foreach my $s (@songs) {  
    say $s if $s =~ /Love/;  
}
```

```
Bad At Love by Halsey  
Love. by Kendrick Lamar Featuring Zacari  
What Lovers Do by Maroon 5 Featuring SZA  
Greatest Love Story by LANCO  
Like I Loved You by Brett Young  
Tell Me You Love Me by Demi Lovato
```

# Meta-Characters

- Meta-Characters are defined by each regex implementation
- They do not match themselves, but instead have special meaning
- We will go into detail about each of them, but some examples are
  - .
  - \
  - [
  - ]
  - (
  - )

## Matching a Single Character

- There are three main ways to match a single character using meta-characters
  - The dot character
  - Programmer-defined character class
  - Built-in character class

# The Dot Character

- Matches any single character **except** newlines
  - This behavior can usually be changed to also match newlines

```
In [38]: foreach my $s (@songs) {  
    say $s if $s =~ /M.n/;  
}
```

```
MotorSport by Migos, Nicki Minaj & Cardi B  
Bodak Yellow (Money Moves) by Cardi B  
I Get The Bag by Gucci Mane Featuring Migos  
Feel It Still by Portugal. The Man  
The Way Life Goes by Lil Uzi Vert Featuring Nicki Minaj  
Unforgettable by French Montana Featuring Swae Lee  
Mine by Bazzi  
Five More Minutes by Scotty McCreery
```

## Character Class

- Often matching any character is not the desired outcome
- To specify a set of characters to match, enclose them in square brackets []
- Characters can either be enumerated like
  - [abcde]
- Or defined as a range by using a hyphen inside the brackets
  - [a-z]

```
In [39]: foreach my $s (@songs) {  
    say $s if $s =~ /W[oa]l/;  
}
```

Wolves by Selena Gomez X Marshmello  
Sky Walker by Miguel Featuring Travis Scott  
You Broke Up With Me by Walker Hayes



## More Character Class Examples

```
In [40]: foreach my $s (@songs) {  
    say $s if $s =~ /[, ']/;  
}
```

God's Plan by Drake

MotorSport by Migos, Nicki Minaj & Cardi B

Keke by 6ix9ine, Fetty Wap & A Boogie Wit da Hoodie

Pills And Automobiles by Chris Brown Featuring Yo Gotti, A Boogie Wit da Hoodie & Kodak Black

King's Dead by Jay Rock, Kendrick Lamar, Future & James Blake

Best Friend by Sofi Tukker Featuring NERVO, The Knocks & Alisa Ueno

The Greatest Show by Hugh Jackman, Keala Settle, Zac Efron, Zendaya & The Greatest Showman Ensemble

I'll Name The Dogs by Blake Shelton

```
In [41]: foreach my $s (@songs) {  
    say $s if $s =~ /[1-9]/;  
}
```

```
Rockstar by Post Malone Featuring 21 Savage  
Gummo by 6ix9ine  
Bartier Cardi by Cardi B Featuring 21 Savage  
What Lovers Do by Maroon 5 Featuring SZA  
Keke by 6ix9ine, Fetty Wap & A Boogie Wit da Hoodie  
1-800-273-8255 by Logic Featuring Alessia Cara & Khalid  
Bank Account by 21 Savage  
Kooda by 6ix9ine  
Wait by Maroon 5
```

## Negation of a Character Class

- By placing a caret symbol (^) as the first character after the bracket, the meaning becomes *match anything but this character class*
- `[^1-9]` matches any character besides a digit

```
In [45]: foreach my $s (@songs) {  
    say $s if $s =~ /Lov[^e]/;  
}
```

Sorry Not Sorry by Demi Lovato

Echame La Culpa by Luis Fonsi & Demi Lovato

Tell Me You Love Me by Demi Lovato

## Built-in Character Classes

- Many specific character classes are used over and over
  - It becomes very time consuming to always type out [1-9] or [a-zA-Z]
- These common classes have shortcuts that can be used to refer to them
  - \w matches all letters and numbers
  - \W matches everything but letters and numbers
  - \d matches all numbers
  - \D matches everything but numbers
  - \s matches any space
  - \S matches any non-space

## Built-in Character Class Examples

```
In [46]: foreach my $s (@songs) {  
    say $s if $s =~ /\d\d/;  
}
```

```
Rockstar by Post Malone Featuring 21 Savage  
Bartier Cardi by Cardi B Featuring 21 Savage  
1-800-273-8255 by Logic Featuring Alessia Cara & Khalid  
Bank Account by 21 Savage
```

```
In [48]: foreach my $s (@songs) {  
    say $s if $s =~ /by\s\d\d/;  
}
```

```
Bank Account by 21 Savage
```

# Character Class Practice

```
In [49]: # Write a regular expression that finds all songs/artists in the top 100 with
# two capital letters right next to each other
foreach my $s (@songs) {
    say $s if $s =~ /[A-Z][A-Z]/;
}
```

Let You Down by NF  
Roll In Peace by Kodak Black Featuring XXXTENTACION  
Lights Down Low by MAX Featuring gnash  
What Lovers Do by Maroon 5 Featuring SZA  
All The Stars by Kendrick Lamar & SZA  
Greatest Love Story by LANCO  
Rubbin Off The Paint by YBN Nahmir  
One Foot by WALK THE MOON  
No Name by NF  
MIC Drop by BTS Featuring Desiigner  
Best Friend by Sofi Tukker Featuring NERVO, The Knocks & Alisa Ueno  
IDGAF by Dua Lipa

```
In [50]: # Write a regular expression that finds all songs/artists in the top 100 with
# a 'word' made up of only non-alphanumeric characters

foreach my $s (@songs) {
    say $s if $s =~ /\s\W\s/;
}
```

Finesse by Bruno Mars & Cardi B  
No Limit by G-Eazy Featuring A Rocky & Cardi B  
MotorSport by Migos, Nicki Minaj & Cardi B  
Meant To Be by Bebe Rexha & Florida Georgia Line

# Alteration

- Sometimes we want to match from a set of not just character, but entire strings
- The pipe character | can be used to indicate alteration
- Important note about ordering: The order of the regex doesn't matter, the first string matched in the text will be used

```
In [51]: foreach my $s (@songs) {  
    say $s if $s =~ /\sYou\s|\sMe\s/;  
}
```

```
Let You Down by NF  
Shape Of You by Ed Sheeran  
Marry Me by Thomas Rhett  
Let Me Go by Hailee Steinfeld & Alesso Featuring Florida Georgia Line & Watt  
Like I Loved You by Brett Young  
This Is Me by Keala Settle & The Greatest Showman Ensemble  
You Broke Up With Me by Walker Hayes  
Tell Me You Love Me by Demi Lovato  
All On Me by Devin Dawson
```

## Grouping

- We will see shortly that it is very useful to group together certain parts of an expression
  - This will also be useful when we talk about substitution
- To group anything together, wrap it in parentheses ( )

```
In [52]: foreach my $s (@songs) {  
    say $s if $s =~ /\s(You|Me)\s/;  
}
```

```
Let You Down by NF  
Shape Of You by Ed Sheeran  
Marry Me by Thomas Rhett  
Let Me Go by Hailee Steinfeld & Alesso Featuring Florida Georgia Line & Watt  
Like I Loved You by Brett Young  
This Is Me by Keala Settle & The Greatest Showman Ensemble  
You Broke Up With Me by Walker Hayes  
Tell Me You Love Me by Demi Lovato  
All On Me by Devin Dawson
```



# Quantifiers

- Quantifiers allow us to specify how many times a particular character, class, or group should occur
- There are 4 main types of quantifiers
  - ? must occur 0 or 1 times
  - \* must occur 0 or more times
  - + must occur 1 or more times
  - Curly braces can be used to provide a custom range

## Quantifier Examples

```
In [53]: foreach my $s (@songs) {  
    say $s if $s =~ /Not?\s/;  
}
```

No Limit by G-Eazy Featuring A Rocky & Cardi B  
Sorry Not Sorry by Demi Lovato  
No Smoke by YoungBoy Never Broke Again  
No Name by NF

```
In [54]: foreach my $s (@songs) {  
    say $s if $s =~ /Lov.*You/;  
}
```

Like I Loved You by Brett Young

# Quantifier Examples

```
In [55]: foreach my $s (@songs) {  
    say $s if $s =~ /St\w+\s/;  
}
```

Feel It Still by Portugal. The Man  
Let Me Go by Hailee Steinfeld & Alesso Featuring Florida Georgia Line & Watt  
Stir Fry by Migos  
All The Stars by Kendrick Lamar & SZA  
Greatest Love Story by LANCO  
Rewrite The Stars by Zac Efron & Zendaya

```
In [56]: foreach my $s (@songs) {  
    say $s if $s =~ /You\w+.*\sby\s/;  
}
```

Young Dumb & Broke by Khalid  
Yours by Russell Dickerson

```
In [62]: foreach my $s (@songs) {  
    say $s if $s =~ /D\w{1,4}g/;  
}
```

Thunder by Imagine Dragons  
Believer by Imagine Dragons  
MIC Drop by BTS Featuring Desiigner  
My Dawg by Lil Baby  
I'll Name The Dogs by Blake Shelton

# Grouping and Quantifiers

- When a quantifier is applied to a group, it means the entire pattern is repeated

```
In [63]: foreach my $s (@songs) {  
    say $s if $s =~ /\s(\w\w\w\w)+\s/;  
}
```

God's Plan by Drake  
Rockstar by Post Malone Featuring 21 Savage  
Finesse by Bruno Mars & Cardi B  
Bad At Love by Halsey  
Diplomatic Immunity by Drake  
Meant To Be by Bebe Rexha & Florida Georgia Line  
Too Good At Goodbyes by Sam Smith  
Let You Down by NF  
Love. by Kendrick Lamar Featuring Zacari  
Gucci Gang by Lil Pump  
I Fall Apart by Post Malone  
How Long by Charlie Puth  
I Get The Bag by Gucci Mane Featuring Migos  
Plain Jane by A Ferg  
End Game by Taylor Swift Featuring Ed Sheeran & Future  
Sorry Not Sorry by Demi Lovato  
Young Dumb & Broke by Khalid  
Never Be The Same by Camila Cabello  
Lights Down Low by MAX Featuring gnash  
Ric Flair Drip by Offset & Metro Boomin  
The Way Life Goes by Lil Uzi Vert Featuring Nicki Minaj  
Let Me Go by Hailee Steinfeld & Alesso Featuring Florida Georgia Line & Watt  
1-800-273-8255 by Logic Featuring Alessia Cara & Khalid  
Unforgettable by French Montana Featuring Swae Lee

## Quantifier Live Example

```
In [73]: # Write a regular expression that finds all songs in
# the top 100 with more than two artist on
# them (don' worry about "featuring" at first
# , but feel free to add it as a challenge)
foreach my $s (@songs) {
    say $s if $s =~ /by\s([\s\w]+[,&]){2,}/;
}
```

MotorSport by Migos, Nicki Minaj & Cardi B

Keke by 6ix9ine, Fetty Wap & A Boogie Wit da Hoodie

Let Me Go by Hailee Steinfeld & Alesso Featuring Florida Georgia Line & Watt

Pills And Automobiles by Chris Brown Featuring Yo Gotti, A Boogie Wit da Hoodie & Kodak Black

King's Dead by Jay Rock, Kendrick Lamar, Future & James Blake

Best Friend by Sofi Tukker Featuring NERVO, The Knocks & Alisa Ueno

The Greatest Show by Hugh Jackman, Keala Settle, Zac Efron, Zendaya & The Greatest Showman Ensemble

## Quantifier Practice

```
In [10]: # Write a regular expression that finds all songs in the top 100 with
# that have both the words "You" and "Me" in the title
foreach my $s (@songs) {
    say $s if $s =~ /(You[\s\w]*Me|Me\s.*You)\s.*by/;
}
```

You Broke Up With Me by Walker Hayes

Tell Me You Love Me by Demi Lovato

## Greedy and Non-Greedy Quantifiers

- By default, all quantifiers will attempt to match as much text as they can
- To change this behaviour, add an extra ? symbol after the quantifier
- The non-greedy quantifiers are:
  - ??
  - \*?
  - +?

## Greedy vs Non-Greedy Example

```
In [11]: foreach my $s (@songs) {  
  if ($s =~ /St.+1/){  
    # $& is the part of the string that matched our pattern  
    say $s . " : " . $&  
  }  
}
```

Feel It Still by Portugal. The Man : Still by Portugal  
Let Me Go by Hailee Steinfeld & Alesso Featuring Florida Georgia Line & Watt :  
Steinfeld & Alesso Featuring Fl  
Broken Halos by Chris Stapleton : Stapl

```
In [12]: foreach my $s (@songs) {  
  if ($s =~ /St.+?1/){  
    say $s . " : " . $&  
  }  
}
```

Feel It Still by Portugal. The Man : Stil  
Let Me Go by Hailee Steinfeld & Alesso Featuring Florida Georgia Line & Watt :  
Steinfel  
Broken Halos by Chris Stapleton : Stapl



# Anchors

- Anchors allow us to specify that the match must start or end the string
  - Caret ^ as the first symbol of a regex forces the match to occur at the beginning of a string
  - Dollar-sign \$ as the last symbol of the regex forces the match to occur at the end of a string

## Anchor Examples (Start)

```
In [13]: foreach my $s (@songs) {  
    say $s if $s =~ /^St\w+/  
}
```

Stir Fry by Migos

```
In [14]: foreach my $s (@songs) {  
    say $s if $s =~ /^You.*/  
}
```

Young Dumb & Broke by Khalid  
Yours by Russell Dickerson  
You Broke Up With Me by Walker Hayes

# Anchor Examples (End)

```
In [15]: foreach my $s (@songs) {  
    say $s if $s =~ /[A-Z]$/;  
}
```

```
Finesse by Bruno Mars & Cardi B  
No Limit by G-Eazy Featuring A Rocky & Cardi B  
MotorSport by Migos, Nicki Minaj & Cardi B  
Let You Down by NF  
Bodak Yellow (Money Moves) by Cardi B  
Roll In Peace by Kodak Black Featuring XXXTENTACION  
What Lovers Do by Maroon 5 Featuring SZA  
All The Stars by Kendrick Lamar & SZA  
Greatest Love Story by LANCO  
One Foot by WALK THE MOON  
La Modelo by Ozuna x Cardi B  
No Name by NF
```

```
In [16]: foreach my $s (@songs) {  
    say $s if $s =~ /s$/;  
}
```

```
Thunder by Imagine Dragons  
I Get The Bag by Gucci Mane Featuring Migos  
Believer by Imagine Dragons  
Stir Fry by Migos  
Sick Boy by The Chainsmokers  
You Broke Up With Me by Walker Hayes  
Rock by Plies
```

# Anchor Practice

```
In [17]: # Write a regular expression that finds all songs in the top 100
# that start with the word 'The'
foreach my $s (@songs) {
    say $s if $s =~ /^The/;
}
```

The Way Life Goes by Lil Uzi Vert Featuring Nicki Minaj  
The Greatest Show by Hugh Jackman, Keala Settle, Zac Efron, Zendaya & The Greatest Showman Ensemble

```
In [19]: # Write a regular expression that finds all songs in the top 100
# only have two words in the title
foreach my $s (@songs) {
    say $s if $s =~ /^S+s\S+sb/;
}
```

God's Plan by Drake  
Diplomatic Immunity by Drake  
New Rules by Dua Lipa  
No Limit by G-Eazy Featuring A Rocky & Cardi B  
Gucci Gang by Lil Pump  
Bartier Cardi by Cardi B Featuring 21 Savage  
How Long by Charlie Puth  
Plain Jane by A Ferg  
Sky Walker by Miguel Featuring Travis Scott  
End Game by Taylor Swift Featuring Ed Sheeran & Future  
Marry Me by Thomas Rhett  
Mi Gente by J Balvin & Willy William Featuring Beyonce  
Bank Account by 21 Savage  
Outside Today by YoungBoy Never Broke Again

## Matching Spaces

- We saw earlier that the `\s` character class matches all spaces
- To match a particular space (very useful in data processing) use the following:
  - `\t` tab character
  - `\n` newline character
  - `\r` carriage return
  - `\f` form feed
- `\b` matches a word boundary, either a space, start or end of a line, etc.

## Boundary Matching is 0-Width

```
In [20]: foreach my $s (@songs) {  
    say $s if $s =~ /by\bCardi B/;  
}
```

```
In [21]: foreach my $s (@songs) {  
    say $s if $s =~ /by\sCardi B/;  
}
```

Bartier Cardi by Cardi B Featuring 21 Savage  
Bodak Yellow (Money Moves) by Cardi B

## Escaping Meta-Characters

- To match a meta-character as a literal, use the backslash to escape it
  - `.` matches a literal period in the string

```
In [22]: foreach my $s (@songs) {  
    say $s if $s =~ /\./;  
}
```

```
Love. by Kendrick Lamar Featuring Zacari  
Feel It Still by Portugal. The Man
```

## Modifiers

- Modifiers are placed after the final delimiter in Perl to change the behavior of the regex
  - Other languages may use flags or arguments to functions
- Common modifiers are
  - `i` - perform a case insensitive match
  - `g` - matches all instances in a string, not just the first
    - important if you want to substitute all matches

```
In [23]: foreach my $s (@songs) {  
         say $s if $s =~ /\bof\b/i;  
}
```

Shape Of You by Ed Sheeran



## Wrap-Up Exercise

```
In [25]: # Write a regular expression that finds all songs in the top 100
# a character besides a alphanumeric or a space in the title
foreach my $s (@songs) {
    say $s if $s =~ /.^[^s\w].*by/;
}
```

God's Plan by Drake

Love. by Kendrick Lamar Featuring Zacari

Him & I by G-Eazy & Halsey

Bodak Yellow (Money Moves) by Cardi B

Young Dumb & Broke by Khalid

1-800-273-8255 by Logic Featuring Alessia Cara & Khalid

King's Dead by Jay Rock, Kendrick Lamar, Future & James Blake

I'll Name The Dogs by Blake Shelton