
# Informe de primera Tarea

Asignatura:
Introducción a la Ciencia de Datos

Grupo 3

Mayo 2024, Montevideo, Uruguay

*Integrantes*
Dominguez, Leandro
Romani, Tiziana

# Indice

# Introducción

El proyecto propuesto abarca un análisis sobre de la obra completa de William Shakespeare, utilizando para ello una base de datos relacional abierta. La tarea implica la conexión a la base de datos y el manejo de los datos a través de Python, específicamente con Pandas en un entorno de Jupyter Notebook.

La primera parte de la tarea se centra en el cargado y la limpieza de los datos. Esto incluye la ejecución de código preestablecido para la importación de datos en DataFrames, la evaluación de la calidad de los datos, y la identificación y corrección de posibles problemas como datos faltantes u otro tipo de errores o inconsistencias. Además, se explorará la distribución de la obra de Shakespeare a lo largo del tiempo a través de gráficos que reflejan la cantidad de obras escritas en diferentes períodos identificando tendencias.

La segunda parte del trabajo se enfoca en el análisis textual de los scripts de las obras. Se requiere la normalización del texto para realizar un conteo efectivo de palabras y la posterior visualización de las palabras más frecuentes en toda la obra de Shakespeare. Además, se explorarán las diferencias entre géneros y personajes, y se identificarán los personajes que más palabras utilizan en sus parlamentos.

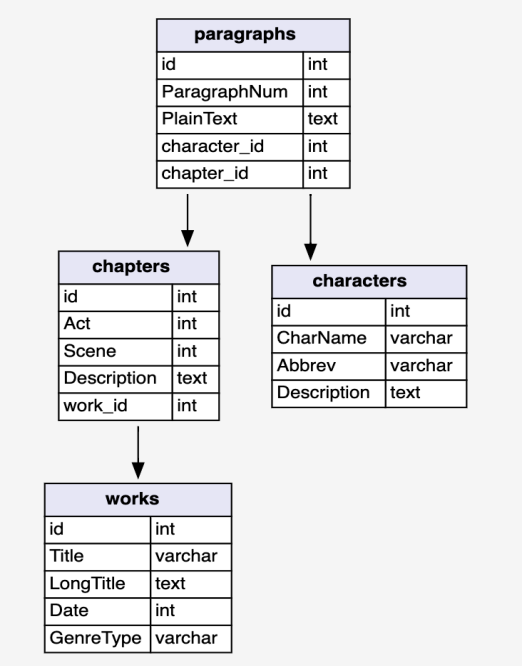
# Cargado y limpieza de datos

Se cargó la base de datos mediante una conexión SQL a una base de datos relacional pública. Esta contiene información acerca de las obras de Shakespeare, particionada en múltiples tablas. Las mismas referencian el contenido de las obras, estructurado en párrafos y capítulos. También referencian entidades o personajes que se asocian con el contenido.

A continuación se incluye la imagen y la descripción del esquema de la BD relacional.

Este esquema cuenta con 4 tablas:

- Paragraphs (párrafos)
- Chapters (capítulos)
- Characters (personajes)
- Works (obras)




**paragraphs:** Almacena los textos de los parlamentos o diálogos, asignados a personajes específicos y situados dentro de un capítulo determinado. Está vinculada tanto a la tabla chapters como a la tabla characters. Los cinco campos que contiene son:

- **id** (int) un identificador de párrafo. Este identificador es único dentro de la tabla aunque eso no se hace presente en la imagen.
- **ParagraphNum** (int) incluye el número de párrafo o diálogo (esto es dentro de un cierto capítulo).
- **PlainText** (text) texto del párrafo.
- **character_id** (int) clave foránea: identificador del personaje que habla en el párrafo. Esto genera una relación con la tabla **chapters**, de esta forma *asignando un personaje a cada párrafo*.
- **chapter_id** (int) clave foránea: incluye el identificador de capítulo al que pertenece el párrafo. Esto genera una relación con la tabla **chapters**, de esta forma *asignando un capítulo a cada párrafo.*

**chapters:** Contiene los detalles de los actos y escenas de cada obra. Está vinculada a la tabla works a través del campo work_id. Los campos incluidos son:

- **id** (int) un identificador de capítulo. Este identificador es único dentro de la tabla aunque eso no se hace presente en el esquema.
- **Act** (int) Número del acto que se identifica con el capítulo.
- **Scene** (int) Número de la escena que se identifica con el capítulo. La combinación de este atributo junto con **Act, y Work** genera una clave primaria asociada a cada capítulo.
- **Description** (text) Descripción del lugar donde sucede la escena.
- **work_id** (int) clave foránea: incluye el identificador de la obra a la que pertenece el capítulo. Esto genera una relación con la tabla **works**, de esta forma *asignando una obra a cada capítulo*.

**characters:** Esta tabla lista todos los personajes que aparecen en las obras de Shakespeare. Incluye los campos:

- **id** (int) un identificador de personaje. Este identificador es único dentro de la tabla aunque no se haga presente en el esquema.
- **CharName** (varchar) nombre de un personaje.
- **Abbrev** (varchar) abreviatura o apodo del personaje.
- **Description** (text) descripción del personaje, en general si es pariente de otro personaje.

**works:** Esta tabla almacena la información general de cada obra de Shakespeare. Cada fila representa una obra distinta e incluye los siguientes campos:

- **id** (int) identificador para las obras. Es único dentro de la tabla aunque no se refleja en el esquema.
- **Title** (varchar) título de la obra.
- **LongTitle** (text) título extendido de la obra.
- **Date** (int) año de publicación de la obra.
- **GenreType** (varchar) género literario de la obra. De entre el conjunto de 5: [Comedy, History, Tragedy, Poem, Sonnet]

De las relaciones se pueden extraer las siguientes conclusiones:

- Una **obra** se compone por **capítulos**.
- Un **capítulo** se compone por **párrafos**.
- Cada **párrafo** se asocia a un **personaje** específico.

# Calidad de datos

## Completitud

En el proceso de revisión de la completitud de los datos, se examina la presencia de valores faltantes en las diferentes tablas. A primera vista, no se observan valores nulos en ninguna de las columnas, considerando que estos representarían la ausencia de datos. Sin embargo, es importante notar que las columnas de tipo texto, que admiten cadenas vacías, requieren un análisis más detallado. En este contexto, una cadena vacía se interpreta como un dato faltante.

En particular, en la tabla characters, se identifica que las columnas **Abbrev** y **Description** contienen cadenas vacías, las cuales se consideran como valores faltantes. Esto subraya la necesidad de tratar estos campos vacíos de manera adecuada durante las fases de limpieza y preprocesamiento de los datos para asegurar la integridad y utilidad del análisis posterior.

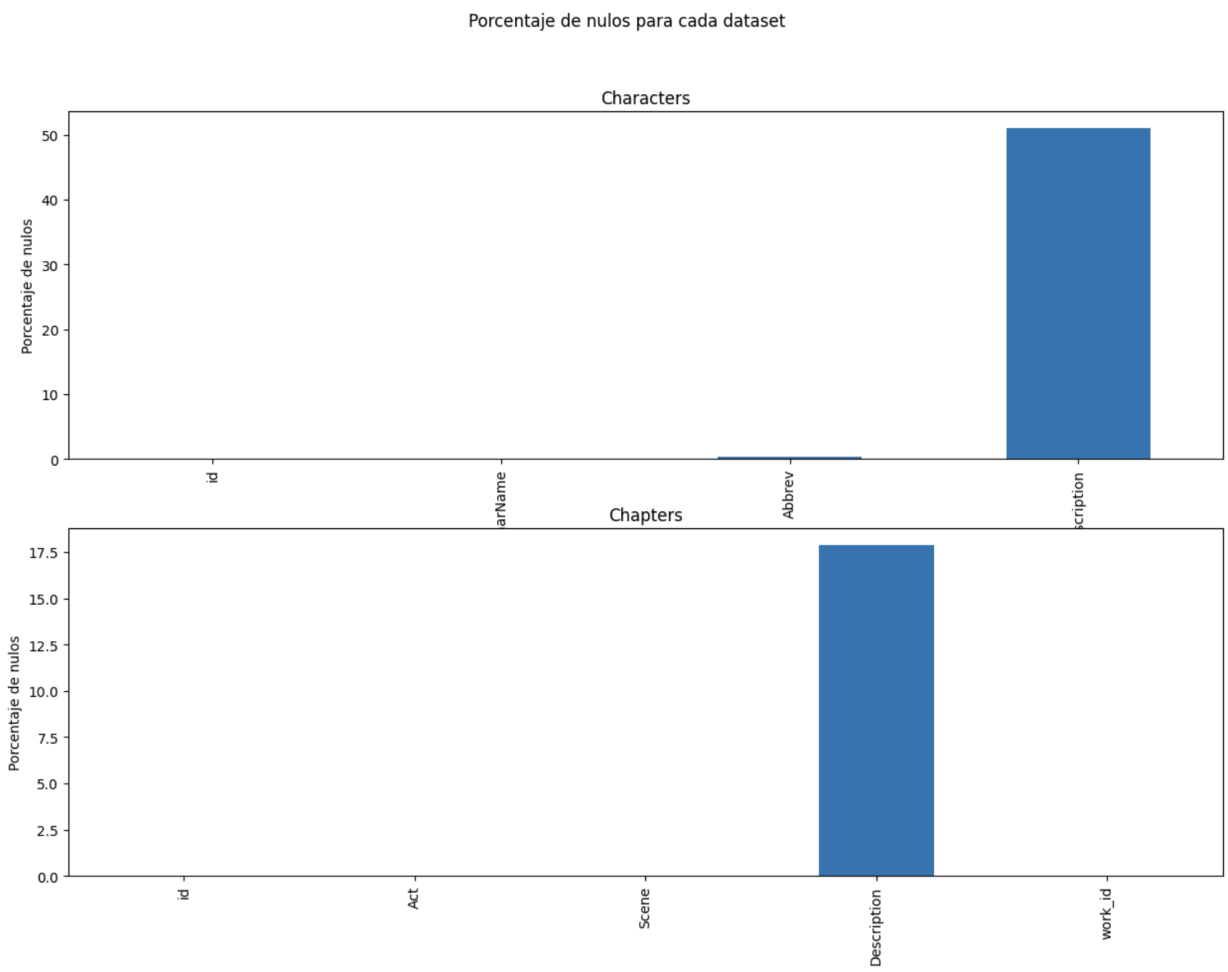
Se hace un pequeño análisis de los personajes con los valores faltantes. Se concluye:

- Un poco más del 50% de los characters tienen la descripción nula
- Un porcentaje muy pequeño no tienen **Abbrev,** pero estos no están contenidos en ningún párrafo
- Aquellos sin descripción (con la excepción del personaje **1261**) son personajes con una cantidad de párrafos reducida, por lo que probablemente sean personajes secundarios.

Algo similar se encuentra en las descripciones de los chapters, donde varios (**169**) tienen una descripción que se considera nula: “---” y “---\n”. Estos representan casi un 18% del total.

Un análisis extensivo de la completitud de los datos, requiere conocimiento del dominio, en este caso todas las obras de Shakespeare. Como quienes redactan no tienen conocimiento del dominio, admiten que puede que existan más faltantes difíciles de identificar.

En la siguiente gráfica pueden verse los porcentajes de nulos dentro de los datasets que contienen datos faltantes:

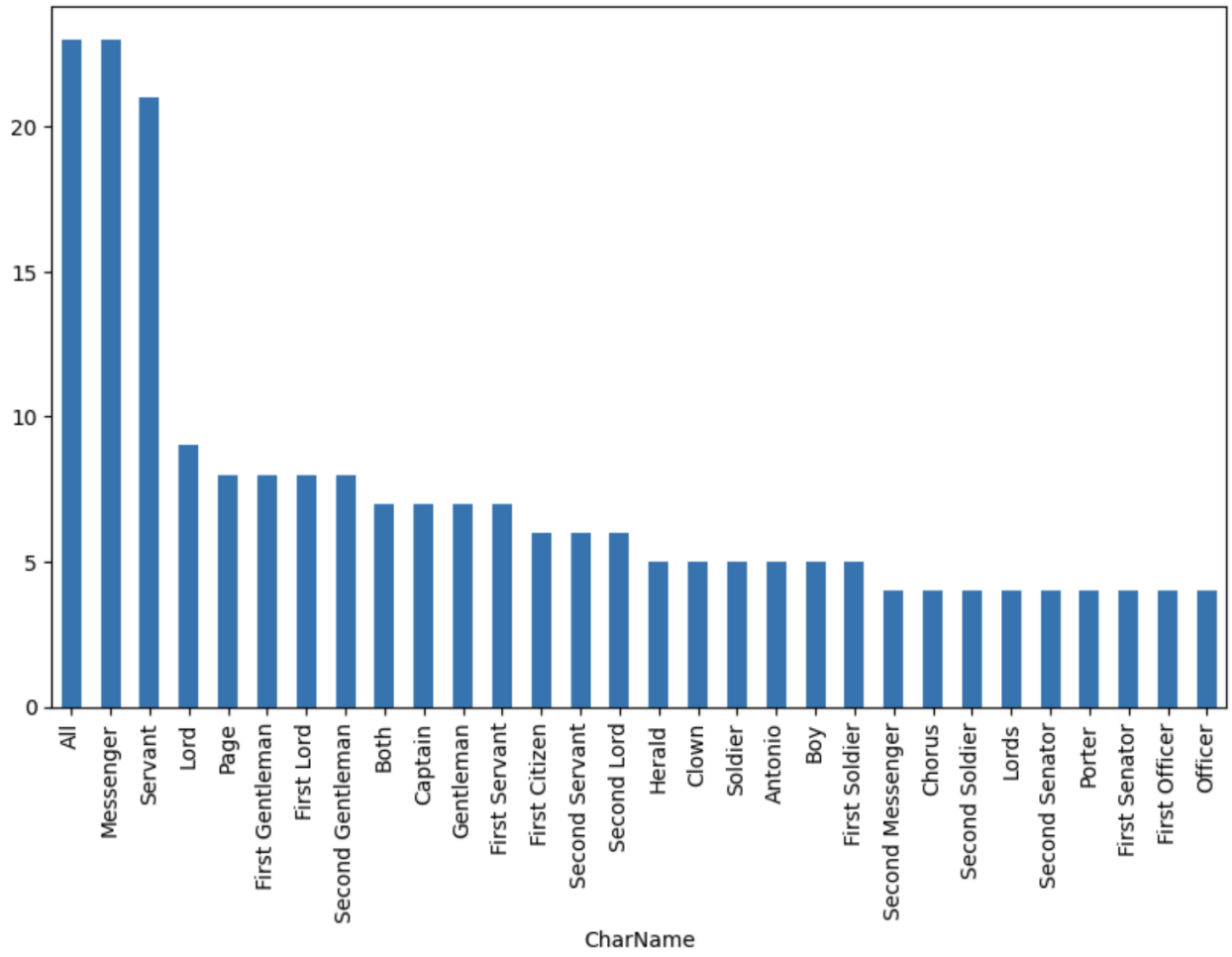



## Exactitud

Para la tabla **works** el atributo **Date** es un *int*. Que en este caso representa un año en el que se creó la obra. Probablemente, dada la época, no hay forma de obtener una fecha más precisa (i.e. con mes por ejemplo)

Al analizar detenidamente la tabla de **personajes**, se observa que algunas entidades listadas pueden no clasificarse estrictamente como personajes en el sentido convencional. En lugar de ello, estas figuras podrían considerarse como voces narrativas o elementos pasivos dentro de las obras. Por ejemplo, la entidad identificada como "Poet", cuya descripción es "the voice of Shakespeare's poetry", no representa un personaje interactivo dentro de la trama, sino más bien la personificación de la voz poética de Shakespeare (o yo lírico). Del mismo modo, las "Stage Directions" funcionan como indicaciones escénicas y no como personajes que contribuyan directamente al desarrollo de la trama.
Si bien hay 1266 personajes distintos, hay 957 personajes con nombre único. Esto se debe a que se repiten nombres en distintas obras. Además de repetirse nombre a lo largo de las obras que son secuelas, también se repiten nombres de personajes genéricos como "sirviente",

"mensajero", "page", etc, que posiblemente sean personajes secundarios o de poca relevancia en las obras. Esto puede verse a continuación:




Por último, con respecto a los párrafos o textos, si uno quisiera hablar de exactitud sintáctica, como “palabras con errores”, o palabras que se dicen diferente hoy en día, se podría hacer un análisis extensivo del vocabulario empleado y cómo se diferencia/relaciona con el vocabulario de hoy. Esto queda fuera del alcance de la tarea.

## Consistencia

Existen varios chequeos que se pueden hacer relacionados a la consistencia de los datos. Entre ellos, chequear que las relaciones tengan sentido, e.g. que no se referencie un personaje en párrafos que no exista. Para efectuar este chequeo es necesario conocer las restricciones del dominio, o en este caso el esquema de la base relacional.

Se chequea cada una de las relaciones definidas en el esquema:

- Si existen ids de personajes en los párrafos para personajes inexistentes en la tabla personajes.
- Si existen capítulos en párrafos que no existan en la tabla capítulos.

- Si existen trabajos asociados a capítulos que no existan en la tabla de trabajos.

Todas se cumplen, por lo que se considera el conjunto de datos consistente en términos de cumplimiento de restricciones de esquema.

Otras consideraciones:

- Se detecta que existe un párrafo (id 1293) sin texto, sin personajes que además no pertenece a ningún capítulo:

| | id | ParagraphNum | PlainText | character_id | chapter_id |
|---|---|---|---|---|---|
| **19580** | 650443 | 1293 | | | |

- Existen 5 personajes que no tienen abreviación y 646 que no tienen descripción. Para los 5 que no tienen abreviación se corroboró que no tienen ningún párrafo asociado

| | id | CharName | Abbrev | Description |
|---|---|---|---|---|
| **559** | 560 | Players | | |
| **633** | 634 | Earl of Kent | | |
| **652** | 653 | John of Lancaster | | son of King Henry IV |
| **1042** | 1043 | Senator | | A senator of Venice |
| **1150** | 1151 | Earl of Surrey | | |

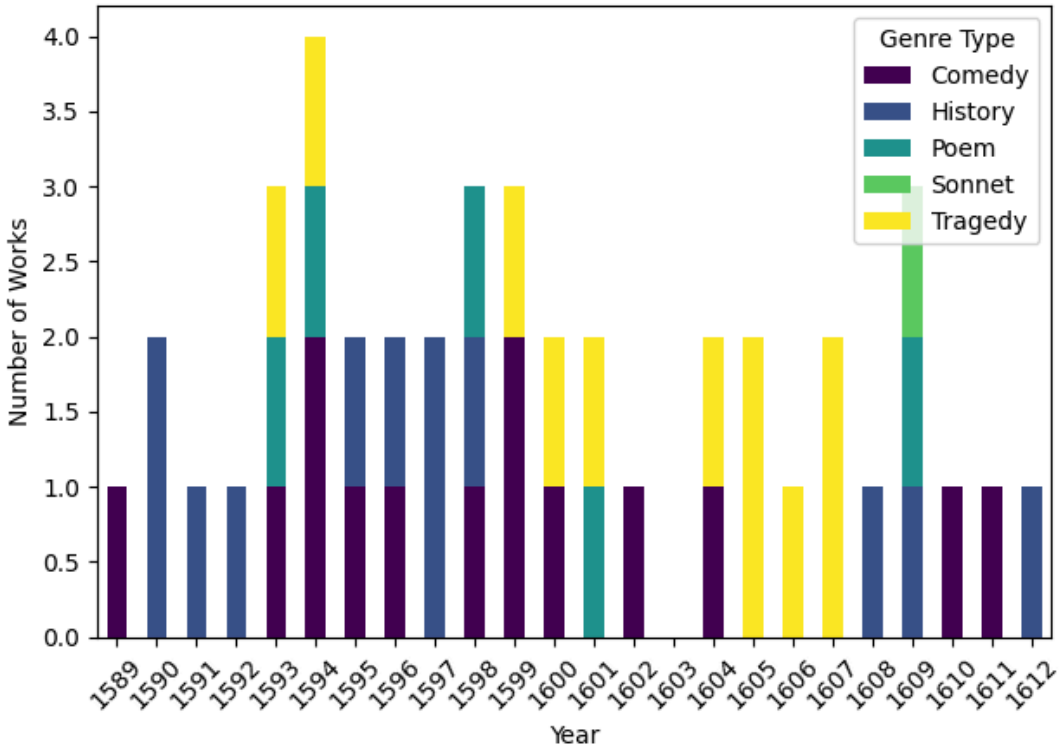
Por otro lado, de los 646 personajes que no tienen descripción, salvo (stage directions), todos tienen muy pocas apariciones con un promedio de 13 párrafos cada uno, por lo que apresuradamente se podría decir que son personajes poco relevantes.

| character_id | |
|---|---|
| 1247.0 | 182 |
| 574.0 | 182 |
| 372.0 | 133 |
| 358.0 | 127 |
| 1238.0 | 121 |
| | ... |
| 842.0 | 1 |
| 1130.0 | 1 |
| 1045.0 | 1 |
| 471.0 | 1 |
| 535.0 | 1 |

# Análisis de datos

## Visualización de obras

La siguiente gráfica se obtiene de evaluar por género las obras realizadas cada año de actividad de Shakespeare.

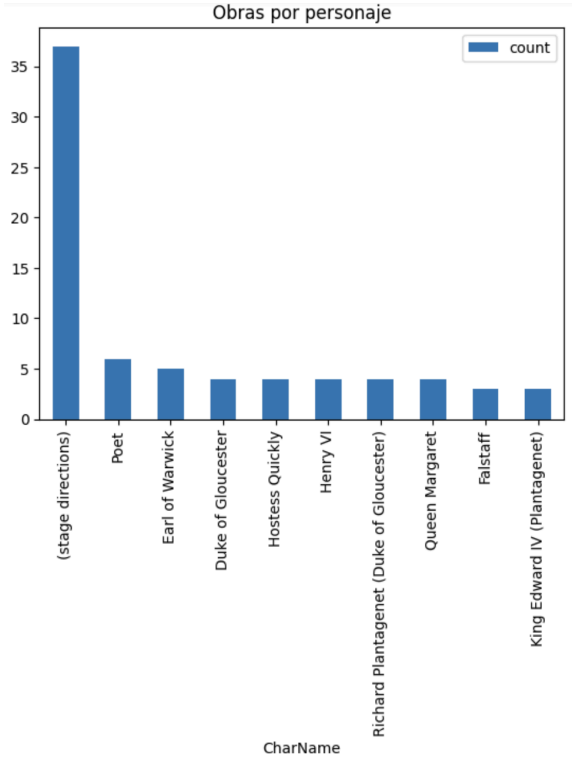


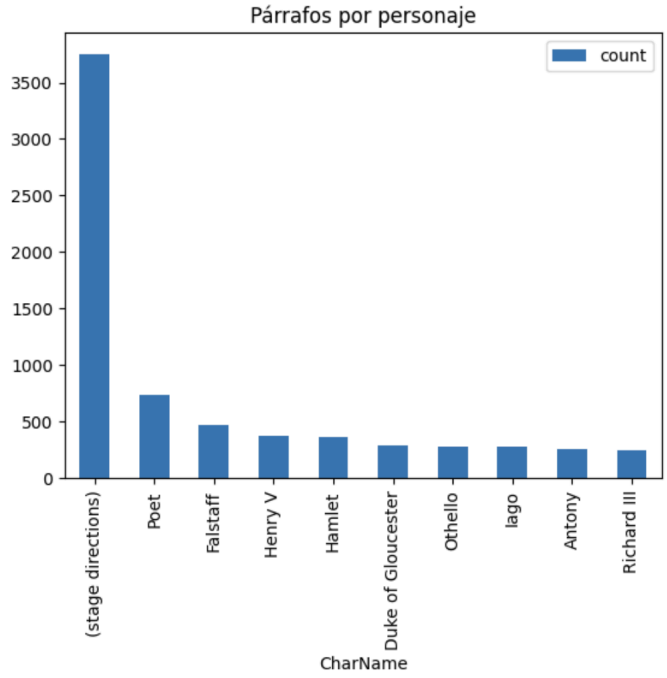

Dada la gráfica, es posible sacar algunas conclusiones:

- Shakespeare tuvo un total de 23 años de actividad. Se mantuvo activo desde 1589 hasta 1612, habiendo un único año en el cual no publicó ningún trabajo: 1603.
- Sus años de mayor actividad son entre 1593 y 1599, donde publicó casi un promedio de 3 obras por año. El resto de los años, publica entre 1 y 2 máximo, con la excepción de 1609 que publica 3 obras.
- Si nos apegamos estrictamente a la estructura de las tablas, su “género favorito” para escribir obras fue *Comedia*. Seguido de *Historia* y *Tragedia* con casi la misma cantidad de obras. A lo largo de su vida escribe algunos poemas y **según la fuente de datos** un único soneto.
  Cabe destacar que, consultando algunas fuentes externas, se sabe que no escribió un único soneto sino que escribió varios a lo largo de su vida y publicó un libro compuesto por 154 de ellos en 1609. Esto es consistente con los 154 párrafos del lo que el conjunto de datos considera un único soneto.
- En general, los géneros se distribuyen de manera homogénea a lo largo de los años, salvo entre 1605 y 1607 que fue claramente un período de tragedia, este puede estar relacionado con la llegada de la peste negra a Londres, que ocurrió entre esas fechas o también relacionarse a algún tipo de crisis personal.

## Párrafos/Obras por personaje

Se podría sacar otro tipo de análisis o conclusiones, por ejemplo la cantidad de obras o de párrafos para cada personaje, ordenando de forma descendente se identifican los personajes más importantes o que tienen más diálogo:




Por ejemplo:

- *(stage directions)* aparece en 37 de las 43 obras. Esto se debe a que aparece en todas las obras salvo los 5 poemas y “el” soneto.
- *Poet* (asociado al yo lírico) aparece en 6 obras, estos se asocian a los 5 poemas y el soneto.
- *Earl of Warwick* es el personaje que aparece en más obras, pero no se encuentra entre los personajes con más párrafos asociados… ¿Será este un personaje de perfil bajo?
- Falstaff parece ser el personaje más charlatán de todos los personajes creados por Shakespeare.

## Conteo de palabras

Como era de esperar, dependiendo de los criterios que se toman para limpiar el texto, será la cantidad de palabras que se considerarán parte de nuestro diccionario base.
Para la limpieza del texto se aplicaron varias técnicas normalmente utilizadas en el Procesamiento de Lenguaje Natural y se comparan los distintos resultados de aplicar o no las mismas.

La función `clean_text` luce de la siguiente forma:

```
def clean_text(df, column_name, stop=False, stem=False):
    # Convertir todo a minúsculas
    result = df[column_name].str.lower()
    # Quitar contracciones
    result = result.apply(transform_contractions)
    # Quitar signos de puntuación y cambiarlos por espacios (" ")
    result = result.apply(remove_punctuation)
    # Quitar stopwords
    if stop:
        result = result.apply(remove_stopwords)
    # Aplicación de stemmer
    if stem:
        result = result.apply(stemmer)

    return result
```

y cuenta principalmente con cinco partes que se describen a continuación:

1) Primeramente se llevan todas las **palabras a minúsculas** para que no haya diferencias entre palabras que ocurren al principio o en el medio de una oración. En general es buena práctica aunque también puede introducir ruido en nombres propios que potencialmente pasan a ser otro tipo de palabras. Ejemplos son: Will, Mark, Grace, entre otros.
2) Otra buena práctica para el idioma inglés es **transformar las contracciones** a su forma expandida de manera que, si en algún lado aparece abreviada y en otras no, se unifiquen criterios. Ejemplos son palabras terminadas en *'t, 'm, 'll*, que se transforman en *not, am* y *will* respectivamente.
3) Como tercer paso, se procede a **eliminar cualquier caracter que no sean números o letras** por medio de expresiones regulares. Esto hace que la palabra "*dijo:*" sea la misma que "*dijo*" o que "*exclamación!* " pase a ser igual que "*exclamación*".
4) Luego de remover todos los signos, se procede a hacer un **filtrado de stopwords**. Las stopwords son palabras que no aportan significado semántico relevante al contexto de una oración y que tienden a tener una alta ocurrencia en los textos. Ejemplos son artículos, preposiciones y pronombres entre otros.

5) Por último se aplica un **proceso de stemming**. Los stemmers son algoritmos que reducen una palabra a una raíz, de forma que distintas conjugaciones de un verbo caen en la misma representación. En nuestro caso se usó el stemmer Porter, que reduce las palabras quitándoles sufijos hasta llegar a una palabra raíz (que no necesariamente tiene que ser una palabra válida en el idioma original). Ejemplos son: *happy* o *happiness* que tendrían *happi* como raíz.

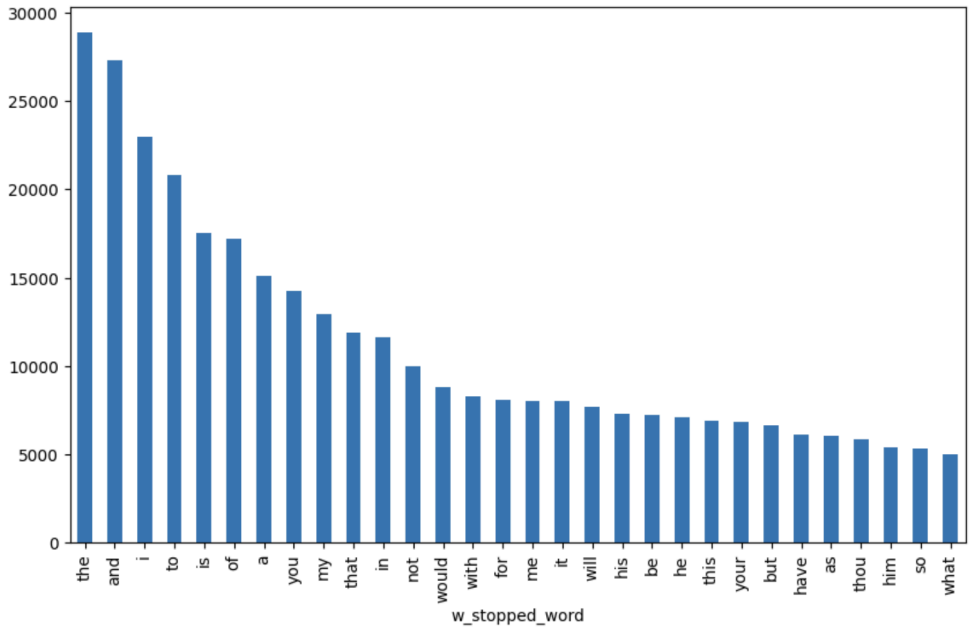
Distintos ejemplos del mismo texto aplicando las diferentes funciones:

| Original | Aplicando 1, 2, 3 | Aplicando 1, 2, 3 y 4 | Aplicando 1, 2, 3, 4 y 5 |
|---|---|---|---|
| [Enter DUKE ORSINO, CURIO, and other Lords; Musicians attending] | enter duke orsino curio and other lords musicians attending | enter duke orsino curio lords musicians attending | enter duke orsino curio lord musician attend |

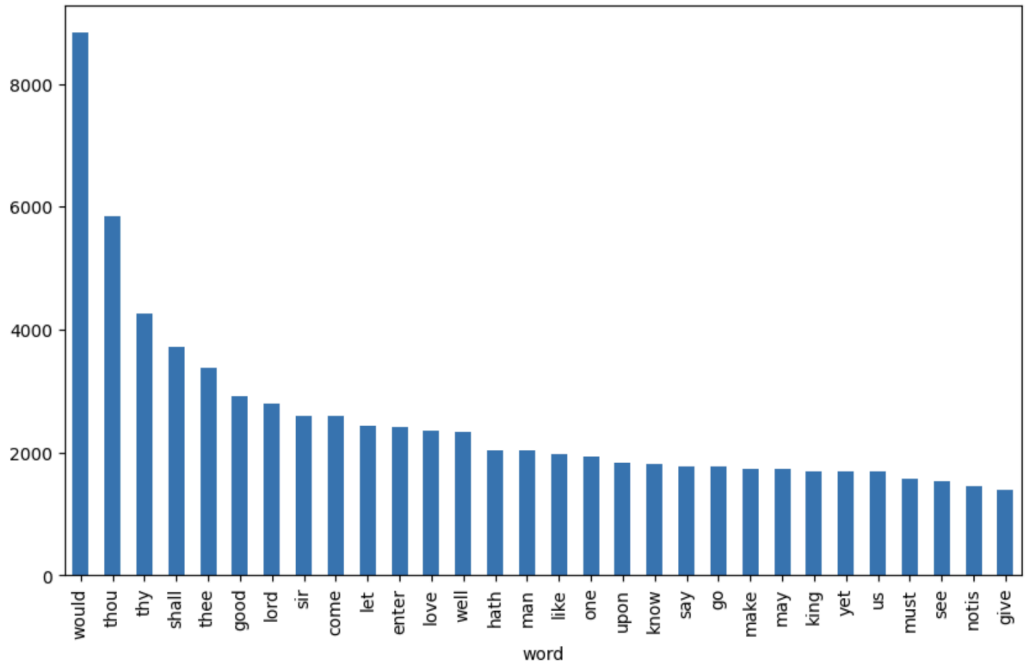
## Palabras más frecuentes según técnica de limpieza

A continuación se muestran y analizan brevemente las 30 palabras más frecuentes según se utilicen las tres posibles combinaciones de la función *clean_text*.

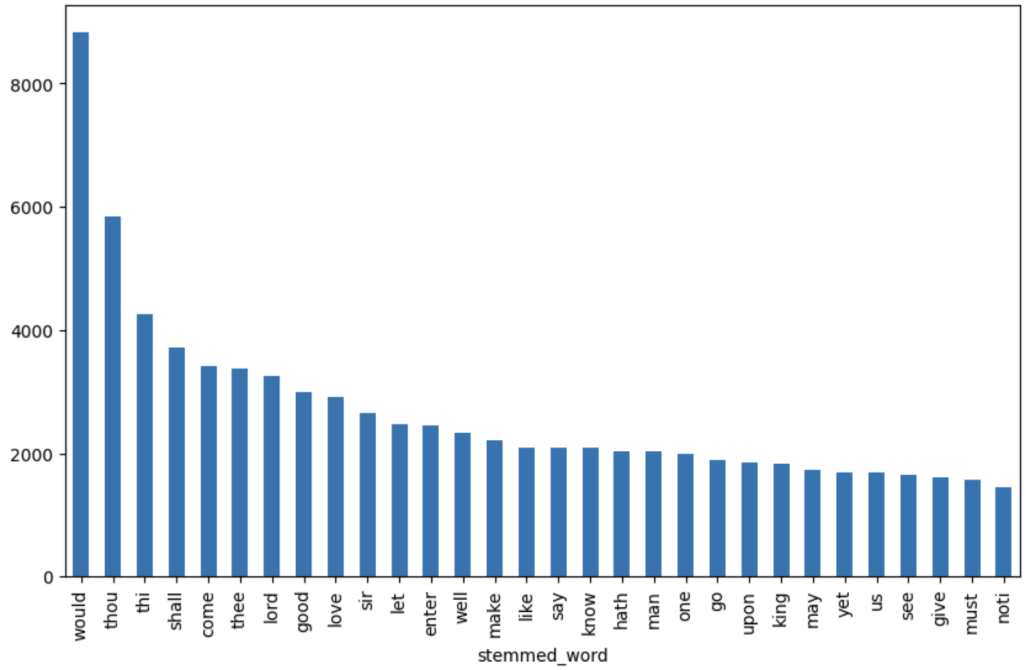
### Ocurrencia de palabras sin sacar stopwords ni aplicando stemmer



Ocurrencia de palabras sacando stopwords, pero no aplicando stemmer




Ocurrencia de palabras sacando stopwords y aplicando stemmer




De la primera gráfica, se puede ver que naturalmente las palabras con mayor frecuencia son las stopwords. Estas palabras no aportan información relevante a la semántica de las oraciones. También podemos ver que aparecen en el orden de decenas de miles, mientras que si filtramos estas palabras, el panorama cambia. A excepción de las palabras *thou*, *thy* y *thee* (que son stopwords pero en inglés antiguo y no estaban incluidas en la lista de stopwords

previstas por la librería NLTK), las palabras más utilizadas comienzan a ser verbos y sustantivos.
Si comparamos las ocurrencias de palabras llevadas a su raíz por medio del stemmer con las que no, podemos notar que (como era de esperar) varios verbos (ej: *come*, *love, make*, *like*, entre otros) suben posiciones en el ranking frente a sustantivos. Esto se debe a que se suman las flexiones de los verbos como si fueran una misma palabra.

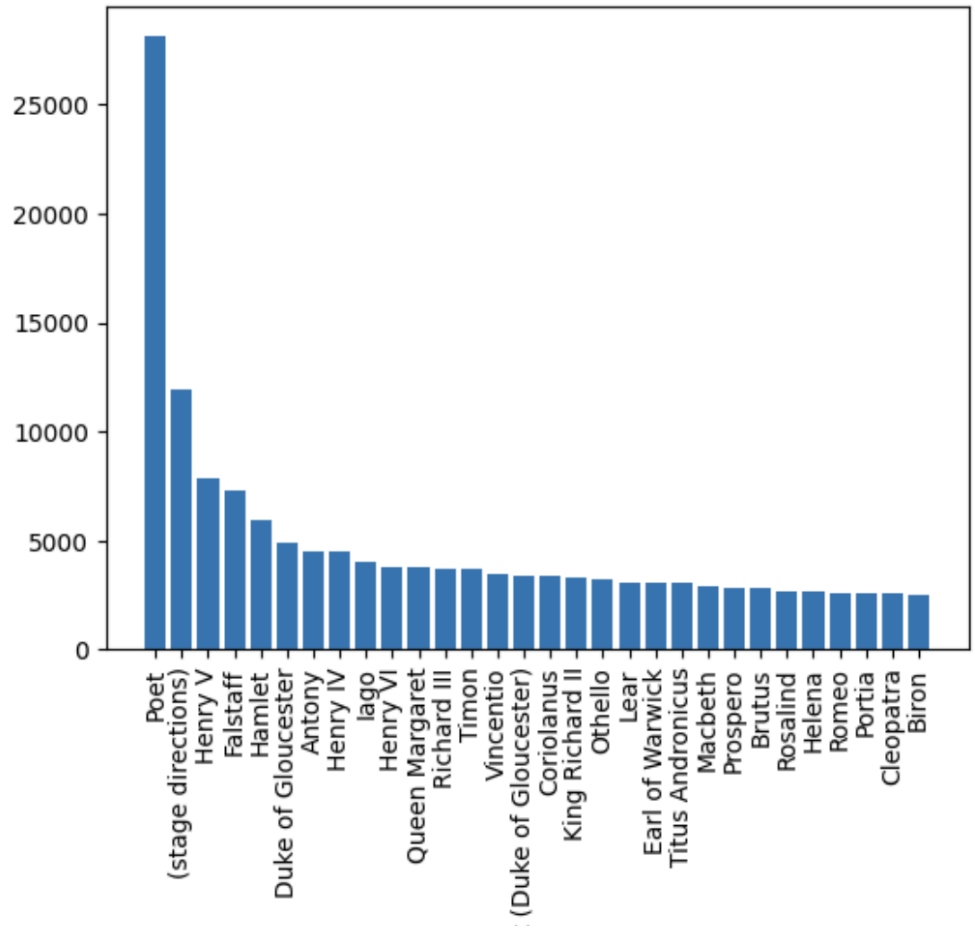
## Otras ideas

Algunas ideas que podrían arrojar resultados interesantes para analizar son:

- **Agrupar palabras por *character_id*:** Esto permitirá identificar las palabras más utilizadas por cada personaje, lo que facilita comparar y determinar si ciertas palabras son distintivas o recurrentes en los diálogos de cada uno. Esto podría ofrecer insights sobre las características o roles de los personajes dentro de las obras.
- **Filtrar y comparar palabras por género:** Al filtrar las obras según su género y cruzar esta información con los datos de capítulos y párrafos, sería posible realizar un análisis de frecuencia de palabras para cada género literario. Esto ayudaría a descubrir cuáles son las palabras más comunes en cada género y podría revelar tendencias o temáticas específicas asociadas a ellos. Existen técnicas como Latent Dirichlet Allocation que permiten distinguir palabras clave entre conjuntos de documentos para saber cuáles son las que caracterizan y distinguen a cada uno del otro.

## Personajes con mayor cantidad de palabras

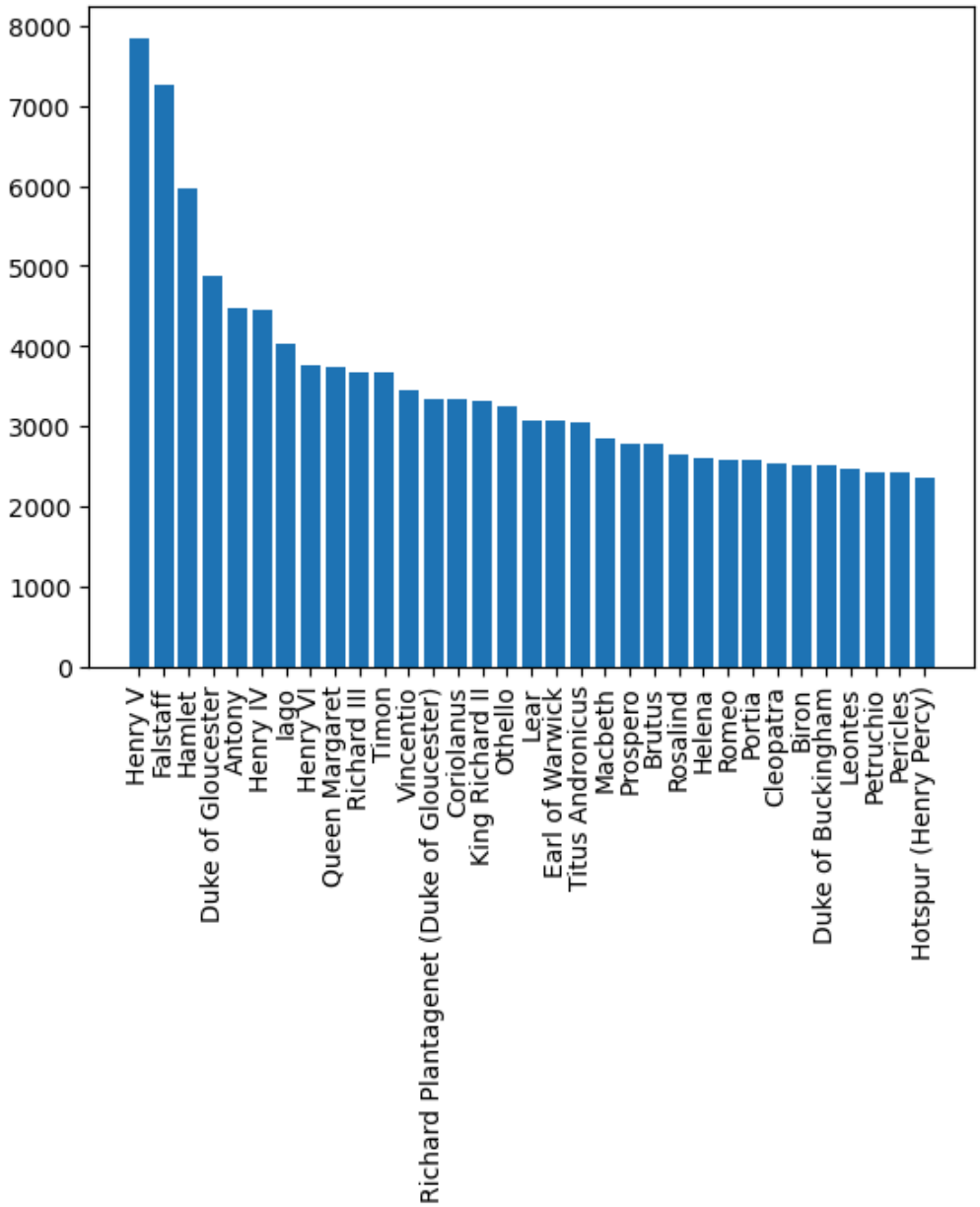
A continuación se observan los 30 personajes con mayor cantidad de palabras habiendo aplicado los correspondientes preprocesamientos de texto.




Como se mencionó previamente, elementos como "Poet" y "Stage Directions" no son personajes en el sentido tradicional, aunque presentan un volumen significativo de palabras. Esto se debe a que cuentan con un número elevado de apariciones en términos de párrafos y presencia en diversas obras, como se observó en los análisis previos de párrafos.

Para abordar este asunto y centrarse en los personajes reales, se puede proceder a filtrar aquellos elementos que no corresponden a personajes propiamente dichos. Una forma práctica de hacerlo es revisar manualmente la lista de personajes; por ejemplo, en el conjunto de datos proporcionado, es evidente que los dos primeros elementos listados como personajes no son reales. Eliminándolos del análisis, podríamos reevaluar y obtener una representación más precisa del uso de palabras por parte de los verdaderos personajes.

Se grafican los personajes con mayor cantidad de palabras excluyendo los mencionados y puede apreciarse que las cantidades de palabras tienden a tener menos varianza.


8000
7000
6000
5000
4000
3000
2000
1000
0
Henry V
Falstaff
Hamlet
Duke of Gloucester
Antony
Henry IV
Iago
Henry VI
Queen Margaret
Richard III
Timon
Vincentio
Richard Plantagenet (Duke of Gloucester)
Coriolanus
King Richard II
Othello
Lear
Earl of Warwick
Titus Andronicus
Macbeth
Prospero
Brutus
Rosalind
Helena
Romeo
Portia
Cleopatra
Biron
Duke of Buckingham
Leontes
Petruchio
Pericles
Hotspur (Henry Percy)

# Posibles preguntas con solución

- Correlación entre personajes. ¿Qué personajes se correlacionan o aparecen en las mismas escenas más seguido?
  - Para resolver esta pregunta, se puede inicializar una tabla de personajes en filas en y columnas. Luego se debería realizar un join de párrafos con capítulos y con personajes. Recorriendo cada escena de cada capítulo, identificar los párrafos asociados y el conjunto de personajes incluídos en estos párrafos y agregar 1 en todas las combinaciones posibles de estos. Por último identificar aquellas combinaciones con los mayores conteos.
- ¿Qué capítulo es aquel que tiene más palabras?
  - Para resolver esta pregunta, se puede realizar un join de las tablas párrafos y capítulos. Luego de limpiar el texto de cada párrafo, contabilizar las palabras separando con un tokenizador. Realizar una agregación por capítulo sumando la cantidad de palabras para cada párrafo y luego ordenar de forma ascendente por este valor.
- ¿Qué personaje aparece en la mayor cantidad de capítulos?
  - Para resolver esto, se puede comenzar realizando un join entre las tablas de párrafos y personajes. Luego se incluyen los capítulos. Por último se agrupa de acuerdo a cada personaje, contabilizando la cantidad de capítulos únicos en los que aparece.
- ¿En qué obra se utiliza el mayor número de palabras únicas?
  - Para resolver esto, se puede comenzar realizando un join entre las tablas capítulos, párrafos y trabajos. Luego se convierte cada fila a un conjunto nuevo de filas basado en el método “.explode” sobre la lista de palabras de cada párrafo. Por último se agrupa de acuerdo a cada obra, contabilizando la cantidad de palabras únicas que aparecen en cada trabajo y se ordenan respectivamente.
- ¿Hay palabras que caracterizan el principio o final de una obra?
  - Para investigar esto, se podría dividir el texto de cada obra en tercios o tomar segmentos representativos y analizar las palabras más frecuentes en cada uno de estos. Esto permitiría identificar patrones en el lenguaje que son característicos de las partes iniciales y finales de las obras. Además, sería interesante comparar las palabras más comunes en las escenas iniciales y finales de cada género, para ver si existen diferencias notables en cómo se introducen o finalizan.
- ¿Qué escena y de qué obra es la que cuenta con la mayor cantidad de personajes presentes al mismo tiempo?
  - Para resolver eso se podría agrupar los datos por obra y por escena, y contar el número de personajes únicos en cada escena. Esto se puede hacer mediante una operación de agrupación en un DataFrame, usando funciones como groupby() seguido de nunique() en Pandas.
- ¿Existe un cambio en las palabras más frecuentes utilizadas en las obras de Shakespeare a lo largo del tiempo?

- Hay varias sospechas de que Shakespeare no fue una única persona sino un conjunto de escritores bajo un pseudónimo. Se podrían contar las palabras características a lo largo de cada obra y ver si hay diferencias significativas entre ellas.