

# Movies

Bartolomé Mestre Fons

Lunes, 21 de Octubre de 2024

---

## Introducción

En este proyecto se desarrolla en Python un análisis básico de datos sobre películas de cine de IMDB. El set de datos que vamos a usar inicialmente se encuentra en la siguiente página:

<https://www.kaggle.com/datasets/carolzhangdc/imdb-5000-movie-dataset>

En ella puede encontrarse información más detallada, así como una descripción precisa de cada columna.

Se plantean los apartados iniciales para su resolución *sin usar la librería pandas*. Esto se propone así para forzar a practicar con las técnicas, herramientas y conceptos básicos de Python. Más adelante, hay otros apartados propuestos para su resolución con esta librería y otras.

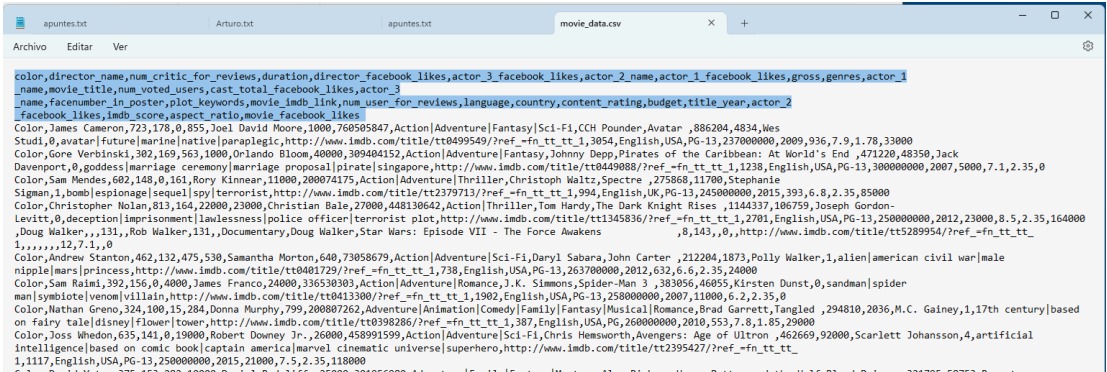
A partir de los datos proporcionados he conseguido realizar con éxito una exploración inicial de los archivos de datos, así como extraer aquellos campos de interés (A.1 y A.2). A la vez, he logradop modificar parte de dichos datos (A.3) y recuperar cierta información básica (A.4). Por otra parte, he demostrado con éxito mis conocimientos en el manejo de diccionarios (B.1 y B.2), así como en el uso de la herramienta defaultdict (B.3 y B.4). A continuación, los datos ha sido graficados con éxito en forma de gráficos de línea (C). Seguidamente, se ha empleado las librerías *BeautifulSoup* y *requests* para realizar un *Web Scrapping* preliminar de las páginas webs de un rango de películas con éxito (todo apartado D). También, *Pandas* es una librería con la que el estudiante se halla familiarizado, logrando todos los objetivos mencionados (E). Por otra parte, las herramientas del método de *MapReduce* se han puesto a prueba satisfactoriamente, obteniendo el resultado esperado (F). Para terminar, el estudiante ha diseñado una script en el apartado G que permite visualizar gráficamente la distribución de presupuestos y de géneros de las películas tratadas a lo largo de todo el documento. Aunque al final de este notebook detallaré la calificación que calculo honestamente, globalmente, siguiendo

las puntuaciones que se asigna a cada apartado, diría que he obtenido una nota de 9.5 sobre 10.

# Datos de partida

(**Nota previa:** hasta el apartado específico de dataframes, se deben desarrollar las soluciones a los ejercicios propuestos sin usar la librería `pandas` , justamente para obligar a practicar con conceptos básicos de Python.)

Nuestra tabla de datos es el archivo de texto `movie_data.csv` que se encuentra en la carpeta `data_in` , y puede verse así con cualquier editor:



images/texto.png La primera fila es la cabecera. Esta fila cabecera contiene los nombres de los campos, separados por comas. Yo la he marcado en azul para distinguirla fácilmente de las demás filas, que contienen los datos propiamente dichos, esto es, los valores de dichos campos, consignando los datos de cada película en cada fila. Si abrimos esta tabla con *excell* (importar datos csv con el separador `,` ), vemos cada dato en una celda.

[Barra de fórmulas]																		
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
color	director_name	num_critic_for_reviews	duration	director_facebook_likes	actor_3_facebook_likes	actor_2_name	actor_1_facebook_likes	gross	genres	actor_1_name	movie_title	num_voted_users	cast_total_facebook_likes	actor_1_facebook_likes	imdb_score	aspect_ratio	movie_facebook_likes	actor_1_facebook_likes
Color	James Cameron	723	178	0	855	Joel David Moore	1800	760585847	Action Adventure Fantasy Sci-Fi	CCH Pounder	Avatar	886204	4834	Wes Studi	7.9	1.78	33000	Avatar
Color	Gore Verbinski	302	169	563	1000	Orlando Bloom	40000	309404152	Action Adventure Fantasy	Johnny Depp	Pirates of the Caribbean: At World's End	471220	48350	Jack Davenport	7.1	2.35	0	Jack Davenport
Color	Sam Mendes	602	148	0	151	Rory Kinnear	11000	200074175	Action Adventure Thriller	Christoph Waltz	Spectre	275868	11700	Stephanie Sigman	7.1	2.35	0	Stephanie Sigman
Color	Christopher Nolan	813	164	22000	23000	Christian Bale	27000	448130642	Action Thriller	Tom Hardy	The Dark Knight Rises	1144337	106759	Joseph Gordon-Levitt	8.5	2.35	164000	Joseph Gordon-Levitt
Color	Doug Walker	131	131	0	0	Rob Walker	131	0	Documentary	Doug Walker	Star Wars: Episode VII - The Force Awakens	8	145	0	8.5	2.35	85000	0
Color	Andrew Stanton	462	132	475	530	Samantha Morton	640	73058679	Action Adventure Sci-Fi	Daryl Sabara	John Carter	212204	1873	Polly Walker	6.8	2.35	24000	alien american civil war male nipple mars princess
Color	Sam Raimi	392	156	0	4000	James Franco	24000	336530303	Action Adventure Romance	J.K. Simmons	Spider-Man 3	383056	46055	Kirsten Dunst	6.2	2.35	0	sandman spider man symbol venom villain
Color	Nathan Greno	324	100	15	204	Donna Murphy	799	200807262	Adventure Animation Comedy Family Fantasy	Brad Garrett	Tangled	294810	2036	M.C. Gai	7.8	1.85	29000	based on fairy tale disney flower tower
Color	Joss Whedon	635	141	0	15000	Robert Downey Jr.	26000	458991599	Action Adventure Sci-Fi	Chris Hemsworth	Avengers: Age of Ultron	462669	92000	Scarlett Johansson	7.3	2.35	118000	based on comic book captain america marvel cinematic universe superhero
Color	David Yates	375	153	282	10000	Daniel Radcliffe	25000	30195690	Adventure Family Fantasy Mystery	Alan Rickman	Harry Potter and the Half-Blood Prince	321795	58753	Rupert Grint	7.9	2.35	0	based on novel wizard magic school young adult
Color	Zack Snyder	673	183	0	2000	Lauren Cohan	15000	300249062	Action Adventure Sci-Fi	Henry Cavill	Batman v Superman: Dawn of Justice	378539	24450	Alan D. Purnim	6.8	2.35	0	based on comic book superhero dc justice league
Color	Byron Singer	434	169	0	903	Marlon Brando	10000	200804087	Action Adventure Sci-Fi	Kevin Spacey	Superman Returns	240296	29991	Frank Langella	6.7	2.35	0	based on comic book superhero dc justice league
Color	Marc Forster	403	106	395	283	Mathieu Amalric	451	168368427	Action Adventure	Giancarlo Giannini	Quantum of Solace	330784	2023	Rory Kinnear	6.8	2.35	0	action he
Color	Gore Verbinski	313	151	563	1000	Orlando Bloom	40000	423032628	Action Adventure Fantasy	Johnny Depp	Pirates of the Caribbean: Dead Man's Chest	522040	44846	Jack Davenport	7.2	2.35	0	based on novel pirate adventure fantasy
Color	Gore Verbinski	450	150	563	1000	Ruth Wilson	40000	8928919	Action Adventure Western	Johnny Depp	The Lone Ranger	181792	45757	Tom Wilkinson	6.8	2.35	0	based on comic book western action adventure
Color	Zack Snyder	733	143	0	740	Christopher Meloni	15000	29121160	Action Adventure Fantasy Sci-Fi	Henry Cavill	Man of Steel	548073	20450	Harry Lennix	6.8	2.35	0	based on comic book superhero dc justice league
Color	Andrew Adamson	258	150	80	201	Pierfrancesco Favino	22000	141614023	Action Adventure Family Fantasy	Peter Dinklage	The Chronicles of Narnia: Prince Caspian	148922	22697	Damian Alcazar	6.8	2.35	0	based on novel fantasy adventure family
Color	Joss Whedon	303	173	0	19000	Robert Downey Jr.	26000	623279547	Action Adventure Sci-Fi	Chris Hemsworth	The Avengers	995415	87697	Scarlett Johansson	8.3	2.35	0	based on comic book superhero mcu action adventure
Color	Rob Marshall	440	136	252	1000	Sam Claflin	40000	241063075	Action Adventure Fantasy	Johnny Depp	Pirates of the Caribbean: On Stranger Tides	270704	34803	Stephen Graham	6.8	2.35	0	based on novel pirate adventure fantasy
Color	Barry Sonnenfeld	451	106	188	718	Michael Stahlberg	10000	179020854	Action Adventure Comedy Family Fantasy Sci-Fi	Will Smith	Men in Black 3	268154	12572	Nicole Scherzinger	6.8	2.35	0	based on comic book superhero action adventure
Color	Peter Jackson	422	164	0	773	Adam Brown	5000	255108370	Adventure Fantasy	Aidan Turner	The Hobbit: The Battle of the Five Armies	354228	9152	James Nesbitt	7.8	2.35	0	based on novel fantasy adventure action adventure
Color	Marc Webb	599	133	464	963	Andrew Garfield	15000	262030603	Action Adventure Fantasy	Emma Stone	The Amazing Spider-Man	451803	28489	Chris Zylka	7.0	2.35	0	based on comic book superhero action adventure
Color	Kidz Scott	343	156	0	738	William Hurt	801	10519792	Action Adventure Drama History	Mark Adly	Robin Hood	211765	3244	Scott Grimes	6.8	2.35	0	based on novel historical action adventure
Color	Peter Jackson	509	186	0	773	Adam Brown	5000	256355534	Adventure Fantasy	Aidan Turner	The Hobbit: The Desolation of Smaug	483540	9152	James Nesbitt	7.8	2.35	0	based on novel fantasy adventure action adventure
Color	Chris Weitz	251	113	129	1000	Eva Green	16000	70083519	Adventure Family Fantasy	Christopher Lee	The Golden Compass	149019	24106	Kristen Scott Thomas	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	Peter Jackson	446	201	0	84	Thomas Kretschmann	6000	218051200	Action Adventure Drama Romance	Naomi Watts	King Kong	316018	7123	Evan Parke	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	James Cameron	315	194	0	794	Kia Wadell	29000	688612082	Drama Romance	Leonardo DiCaprio	Titanic	789309	45223	Gloria Stuart	9.3	1.37	0	based on novel historical drama romance
Color	Anthony Russo	516	147	94	11000	Scarlett Johansson	21000	40719292	Action Adventure Sci-Fi	Robert Downey Jr.	Captain America: Civil War	272670	64798	Chris Evans	7.7	2.35	0	based on comic book superhero mcu action adventure
Color	Peter Berg	377	131	532	627	Alexander Skarsgård	14000	65173189	Action Adventure Sci-Fi Thriller	Liam Neeson	Battleship	220382	26679	Tadanobu Asano	6.8	2.35	0	based on comic book superhero action adventure
Color	Colin Trevorrow	644	124	363	1000	Judy Greer	3000	62177271	Action Adventure Sci-Fi Thriller	Byrne Dallas Howard	Jurassic World	418214	6438	Omari Sy	7.0	2.35	0	based on novel fantasy adventure action adventure
Color	Sam Mendes	760	143	0	393	Heian McNulty	883	304560277	Action Adventure Thriller	Albert Finney	Spyglass	522030	2038	Rory Kinnear	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	Sam Raimi	300	135	0	4000	James Franco	24000	373277893	Action Adventure Fantasy Romance	J.K. Simmons	Spider-Man 2	411164	45388	Kirsten Dunst	7.2	2.35	0	based on comic book superhero action adventure
Color	Shane Black	608	195	1000	3000	Jon Favreau	21000	408992272	Action Adventure Sci-Fi	Robert Downey Jr.	Iron Man 3	557489	39426	Dan Chevalle	7.7	2.35	0	based on comic book superhero action adventure
Color	Tim Burton	451	109	13000	11000	Alan Rickman	40000	31416266	Adventure Family Fantasy	Johnny Depp	Alice in Wonderland	396230	79507	Anne Hathaway	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	Brett Ratner	334	104	420	560	Kelsey Grammer	20000	234560014	Action Adventure Fantasy Sci-Fi Thriller	Hugh Jackman	X-Men: The Last Stand	385427	21714	Daniel Cuddeback	6.8	2.35	0	based on comic book superhero action adventure
Color	Dan Scanlon	376	104	37	760	Tyler Labine	12000	26488329	Adventure Animation Comedy Family Fantasy	Buena Vista	Monsters University	235025	14863	Sean Hayes	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	Michael Bay	366	150	0	454	Kevin Dunn	894	402076889	Action Adventure Sci-Fi	Glenn Morshower	Transformers: Revenge of the Fallen	323207	3218	Ramon Rodriguez	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	Michael Bay	378	165	0	800	Sophia Miles	274	245428127	Action Adventure Sci-Fi	Bingbing Li	Transformers: Age of Extinction	242420	3980	Kelsey Grammer	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	Sam Raimi	525	130	0	11000	Mila Kunis	44000	234903076	Adventure Family Fantasy	Tim Holmes	On the Beach	175409	28441	James Franco	6.8	2.35	0	based on novel fantasy adventure action adventure
Color	Marc Webb	495	142	464	825	Andrew Garfield	15000	202853933	Action Adventure Fantasy Sci-Fi	Emma Stone	The Amazing Spider-Man 2	321227	26631	B.J. Novak	7.0	2.35	0	based on comic book superhero action adventure
Color	Joseph Kosinski	469	125	364	1000	Olivia Wilde	12000	172051787	Action Adventure Sci-Fi	Jeff Bridges	TRON: Legacy	264183	25500	James Frain	6.8	2.35	0	based on novel fantasy adventure action adventure

Mostramos la hoja excell en dos imágenes por ser muy ancha.

	G	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB
1	actor_3_name	face	plot_keywords	movie_imdb_link	num_votes	language	count	content	budget	title_year	actor_2	imdb	year	movie
2	Wei Studi	0	asterisk [feature] marine [native] paragonic	http://www.imdb.com/title/tt0320540/?ref=fn_1_1	3054	English	USA	PG-13	23700000	2009	536	79	176	33000
3	Jack Davenport	0	goddess [marriage ceremony] [marriage proposal] [pirate] [singapore]	http://www.imdb.com/title/tt0440980/?ref=fn_1_1	1238	English	USA	PG-13	30000000	2007	5000	71	235	0
4	Stephanie Sigman	1	bomb [espionage] sequel [spy] [terrorist]	http://www.imdb.com/title/tt2379713/?ref=fn_1_1	994	English	UK	PG-13	24500000	2015	393	68	235	85000
5	Joseph Gordon-Levitt	0	deception [imprisonment] [lawlessness] [police officer] [terrorist plot]	http://www.imdb.com/title/tt1345836/?ref=fn_1_1	2701	English	USA	PG-13	25000000	2012	23000	85	235	164000
6		0		http://www.imdb.com/title/tt0289654/?ref=fn_1_1								12	71	0
7	Poly Walker	1	alien [american civil war] [male nipple] [mars] [princess]	http://www.imdb.com/title/tt0401729/?ref=fn_1_1	738	English	USA	PG-13	26370000	2012	632	66	235	24000
8	Kirsten Dunst	0	sandman [spider man] [symbiote] [venom] [villain]	http://www.imdb.com/title/tt0141300/?ref=fn_1_1	1902	English	USA	PG-13	25800000	2007	11000	62	235	0
9	M. C. Gainey	1	17th century [based on fairy tale] [disney] [flower] [tower]	http://www.imdb.com/title/tt0398286/?ref=fn_1_1	387	English	USA	PG	26000000	2010	553	78	185	29000
10	Scarlett Johansson	4	artificial intelligence [based on comic book] [captain america] [marvel cinematic universe] [superhero]	http://www.imdb.com/title/tt0395427/?ref=fn_1_1	1117	English	USA	PG-13	25000000	2015	21000	75	235	118000
11	Rupert Grint	3	blood [book] [love] [potion] [professor]	http://www.imdb.com/title/tt0411741/?ref=fn_1_1	973	English	UK	PG	25000000	2009	11000	75	235	10000
12	Alan D. Purwin	0	based on comic book [batman] [sequel to a reboot] [superhero] [superman]	http://www.imdb.com/title/tt2975590/?ref=fn_1_1	3018	English	USA	PG-13	25000000	2016	4000	69	235	197000
13	Frank Langella	0	crystal [epic] [lex luthor] [lois lane] [return to earth]	http://www.imdb.com/title/tt0343150/?ref=fn_1_1	2367	English	USA	PG-13	20900000	2006	10000	61	235	0
14	Rory Kinnear	1	action hero [attempted rape] [bond girl] [official james bond series] [revenge]	http://www.imdb.com/title/tt0839315/?ref=fn_1_1	1243	English	UK	PG-13	20000000	2008	412	67	235	0
15	Jack Davenport	2	box office hit [giant squid] [heart] [lar's dice] [monster]	http://www.imdb.com/title/tt0383744/?ref=fn_1_1	1832	English	USA	PG-13	22500000	2006	5000	73	235	5000
16	Tom Wilkinson	1	horse [outlaw] [texas] [texas ranger] [train]	http://www.imdb.com/title/tt1210819/?ref=fn_1_1	711	English	USA	PG-13	21500000	2013	2000	65	235	48000
17	Harry Lennix	0	based on comic book [british actor playing american character] [final battle] [origin of hero] [revenge]	http://www.imdb.com/title/tt0770828/?ref=fn_1_1	2536	English	USA	PG	22500000	2013	3000	72	235	118000
18	Damian Alcazar	4	brother brother relationship [brother sister relationship] [good versus evil] [king] [narnia]	http://www.imdb.com/title/tt0494448/?ref=fn_1_1	438	English	USA	PG	22500000	2008	216	66	235	0
19	Scarlett Johansson	3	alien invasion [assassin] [battle] [iron man] [soldier]	http://www.imdb.com/title/tt0848228/?ref=fn_1_1	1722	English	USA	PG-13	22000000	2012	21000	81	185	123000
20	Stephen Graham	4	blackbeard [captain] [pirate] [revenge] [soldier]	http://www.imdb.com/title/tt1296500/?ref=fn_1_1	484	English	USA	PG-13	25000000	2011	11000	67	235	58000
21	Nicole Scherzinger	1	alien [criminal] [m.l.b.] [maximum security prison] [prison]	http://www.imdb.com/title/tt1400024/?ref=fn_1_1	341	English	USA	PG-13	22500000	2012	816	68	185	40000
22	James Nesbitt	0	army [elf] [hobbit] [middle earth] [orc]	http://www.imdb.com/title/tt0310332/?ref=fn_1_1	802	English	New Zeal	PG-13	25000000	2014	972	75	235	65000
23	Chris Zylka	0	island [outcast] [spider man] [teenager]	http://www.imdb.com/title/tt0904470/?ref=fn_1_1	1223	English	USA	PG-13	23000000	2012	10000	70	235	56000
24	Scott Grimes	0	1190s [archer] [england] [king of england] [robin hood]	http://www.imdb.com/title/tt0955308/?ref=fn_1_1	546	English	USA	PG-13	20000000	2010	882	67	235	17000
25	James Nesbitt	6	dwarf [elf] [lake town] [mountain] [sword and sorcery]	http://www.imdb.com/title/tt170358/?ref=fn_1_1	951	English	USA	PG-13	22500000	2013	972	79	235	83000
26	Kristin Scott Thomas	2	children [epic] [friend] [girl] [quest]	http://www.imdb.com/title/tt0385752/?ref=fn_1_1	666	English	USA	PG-13	18000000	2007	6000	61	235	0
27	Fon Peke	0	animal name [title] [ape] [adultery a woman] [gorilla] [island] [king kong]	http://www.imdb.com/title/tt0906071/?ref=fn_1_1	2619	English	New Zeal	PG-13	20700000	2005	919	72	235	0
28	Gloria Stuart	0	artist [love] [ship] [titanic] [war]	http://www.imdb.com/title/tt0120338/?ref=fn_1_1	2528	English	USA	PG-13	20000000	1997	14000	77	235	26000
29	Chris Evans	0	based on comic book [knife] [marvel cinematic universe] [returning character killed off] [super]	http://www.imdb.com/title/tt0349820/?ref=fn_1_1	1022	English	USA	PG-13	25000000	2016	19000	82	235	72000
30	Tadanobu Asano	0	box office flop [hawaii] [naval] [oahu hawaii] [ship]	http://www.imdb.com/title/tt1440129/?ref=fn_1_1	751	English	USA	PG-13	20900000	2012	10000	59	235	44000
31	Omni Sy	0	dinosaur [disaster film] [experiment] [gone wrong] [jurassic park] [velociraptor]	http://www.imdb.com/title/tt0308610/?ref=fn_1_1	1290	English	USA	PG-13	15000000	2015	2000	70	20	150000
32	Rory Kinnear	0	brave [childhood home] [computer hacker] [intelligence agency] [terrorist cell]	http://www.imdb.com/title/tt1074638/?ref=fn_1_1	1498	English	UK	PG-13	20000000	2012	563	78	235	80000
33	Kirsten Dunst	1	death [doctor] [scientist] [super villain] [tentacle]	http://www.imdb.com/title/tt0310654/?ref=fn_1_1	1303	English	USA	PG-13	20000000	2004	11000	73	235	0
34	Don Cheadle	3	armor [explosion] [human bomb] [missile attack] [terrorist]	http://www.imdb.com/title/tt1300854/?ref=fn_1_1	1187	English	USA	PG-13	20000000	2013	4000	72	235	95000
35	Anne Hathaway	0	alice in wonderland [mistaking reality for dream] [queen] [shrinking] [shrinking potion]	http://www.imdb.com/title/tt1014759/?ref=fn_1_1	738	English	USA	PG	20000000	2010	25000	65	185	24000
36	Daniel Cudmore	0	battle [musical] [outrage] [walking through a wall] [women]	http://www.imdb.com/title/tt0370944/?ref=fn_1_1	1912	English	Canada	PG-13	21000000	2006	808	68	235	0
37	Sean Hayes	0	cheating [fraternity] [monster] [singing in a car] [university]	http://www.imdb.com/title/tt1453969/?ref=fn_1_1	263	English	USA	G	20000000	2013	779	73	185	44000
38	Ramon Rodriguez	0	autobot [deception] [machine] [sun] [symbol]	http://www.imdb.com/title/tt1055369/?ref=fn_1_1	1439	English	USA	PG-13	20000000	2009	581	60	235	0
39	Kelsey Grammer	2	blockbuster [bumblebee the character] [semi truck and trailer] [texas] [truck]	http://www.imdb.com/title/tt109248/?ref=fn_1_1	919	English	USA	PG-13	21000000	2014	906	57	235	56000
40	James Franco	4	circus [magic] [magician] [old witch]	http://www.imdb.com/title/tt1623290/?ref=fn_1_1	511	English	USA	PG	21500000	2013	15000	64	235	69000
41	B.J. Novak	0	costumed hero [death of girlfriend] [masked vigilante] [returning character killed off] [teenager]	http://www.imdb.com/title/tt1872181/?ref=fn_1_1	1067	English	USA	PG-13	20000000	2014	10000	67	235	41000
42	James Frain	0	arcade [bridge] [disappearance] [escape] [warrior]	http://www.imdb.com/title/tt1100017/?ref=fn_1_1	660	English	USA	PG	17000000	2010	10000	68	235	30000

# Librerías y constantes globales

Pongamos todas las librerías necesarias al principio, tal como propone el estilo  `pep-8` .

Ej.:  `PEP 8 -- Style Guide for Python Code.`

De paso, en éste y otros lugares de Internet podrás encontrar detalles sobre estilo y presentación de código con un estilo estándar y cuidado. Examina esta página y trata de seguir esas indicaciones.

In [3]:

```
# Librerías:
import csv
import time
import requests

from collections import Counter
from collections import defaultdict
from bs4 import BeautifulSoup
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [4]:

```
# Constantes globales:
MOVIES_DATA = "./movie_data.csv" # Se añade el './' delante para poder leerse.

COLUMNS_KEEP = [
    'movie_title', 'title_year', 'director_name', 'actor_1_name',
    'language', 'country', 'color', 'budget', 'imdb_score', 'movie_imdb_link'
] # Estas son las columnas que queremos mantener en el apartado A.2.

FEW_FIELDS = "algunos_campos.txt" # Aquí se guardan los datos en B.2.

HEADERS = {
    "User-Agent": (
        "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:98.0) "
        "Gecko/20100101 Firefox/98.0"
    ),
    "Accept": (
        "text/html,application/xhtml+xml,application/xml;q=0.9,"
```

```

        "image/avif,image/webp,*/*;q=0.8"
    ),
    "Accept-Language": "en-US,en;q=0.5",
    "Accept-Encoding": "gzip, deflate",
    "Connection": "keep-alive",
    "Upgrade-Insecure-Requests": "1",
    "Sec-Fetch-Dest": "document",
    "Sec-Fetch-Mode": "navigate",
    "Sec-Fetch-Site": "none",
    "Sec-Fetch-User": "?1",
    "Cache-Control": "max-age=0",
} # Este es el valor del header a la hora de Leer páginas web (D).

FIELDS_SELECTED = [
    'movie_title', 'color', 'director_name', 'language', 'country',
    'actor_1_name', 'movie_imdb_link'
] # Estos son Los campos que mantendremos en nuestra data en el apartado E2.

BINS = 40 # Este es el número de 'bins' de nuestro diagrama de barras (G)
LOWER_PERCENT = 0.25 # Este es el límite percentual inferior para el apartado G
UPPER_PERCENT = 0.75 # Este es el límite percentual superior para el apartado G.

```

## Parte A. Ejercicios básicos sin usar pandas [2 puntos]

Esta parte inicial debe realizarse sin usar la librería `pandas`. Para practicar con esta librería, se plantean otros apartados más abajo.

### A.1. Exploración inicial básica del archivo de datos

Deseamos cargar el archivo de datos, que tiene un formato `csv`. En este apartado, te recomiendo fuertemente usar la librería `cvs`, que deberás importar en la primera celda del script, más arriba, no aquí. (En los siguientes apartados, ya no mencionaré qué librerías usar ni recordaré dónde se han de importar.)

Observa también que el test de funcionamiento te da el nombre de la función que deber definir y algún otro identificador como es, por ejemplo, la constante `MOVIES_DATA`, que debes definir también más arriba, en la segunda celda de este script.

Finalmente, observando el test, verás que se carga por separado la cabecera y las filas de datos.

```

In [5]: def load_full_data(url: str):
        """
        Carga los datos desde un archivo CSV y devuelve la cabecera y la data
        separadamente.

        Parameters:
        -----
        url : str
            La ruta del archivo o la URL del archivo CSV.

```

```

Returns:
-----
tuple
    Una tupla donde el primer elemento es la cabecer (lista) y el segundo
    elemento son los datos (lista de filas).
"""
# Es aquí importante especificar el formato correcto del encoding.
with open(url, encoding = 'Latin1') as file:
    # El resultado es un iterador, entonces hay que convertirlo a una lista.
    csv_reader = csv.reader(file)

    header = next(csv_reader)
    data = [row for row in csv_reader]

return header, data

```

In [6]: *# Test de funcionamiento*

```

full_header, full_list_data = load_full_data(MOVIES_DATA)

print(full_header)
print()
print(full_list_data[0:5])

```

```

['color', 'director_name', 'num_critic_for_reviews', 'duration', 'director_facebo
ok_likes', 'actor_3_facebook_likes', 'actor_2_name', 'actor_1_facebook_likes', 'g
ross', 'genres', 'actor_1_name', 'movie_title', 'num_voted_users', 'cast_total_fa
cebook_likes', 'actor_3_name', 'facenumber_in_poster', 'plot_keywords', 'movie_im
db_link', 'num_user_for_reviews', 'language', 'country', 'content_rating', 'budge
t', 'title_year', 'actor_2_facebook_likes', 'imdb_score', 'aspect_ratio', 'movie_
facebook_likes']

```

```

[['Color', 'James Cameron', '723', '178', '0', '855', 'Joel David Moore', '1000',
'760505847', 'Action|Adventure|Fantasy|Sci-Fi', 'CCH Pounder', 'Avatar\u00c2\u00e0
6204', '4834', 'Wes Studi', '0', 'avatar|future|marine|native|paraplegic', 'htt
p://www.imdb.com/title/tt0499549/?ref=fn_tt_tt_1', '3054', 'English', 'USA', 'PG
-13', '237000000', '2009', '936', '7.9', '1.78', '33000'], ['Color', 'Gore Verbin
ski', '302', '169', '563', '1000', 'Orlando Bloom', '40000', '309404152', 'Action
|Adventure|Fantasy', 'Johnny Depp', 'Pirates of the Caribbean: At World's End\u00c2
0', '471220', '48350', 'Jack Davenport', '0', 'goddess|marriage ceremony|marriage
proposal|pirate|singapore', 'http://www.imdb.com/title/tt0449088/?ref=fn_tt_tt_
1', '1238', 'English', 'USA', 'PG-13', '300000000', '2007', '5000', '7.1', '2.3
5', '0'], ['Color', 'Sam Mendes', '602', '148', '0', '161', 'Rory Kinnear', '1100
0', '200074175', 'Action|Adventure|Thriller', 'Christoph Waltz', 'Spectre\u00c2
275868', '11700', 'Stephanie Sigman', '1', 'bomb|espionage|sequel|spy|terroris
t', 'http://www.imdb.com/title/tt2379713/?ref=fn_tt_tt_1', '994', 'English', 'U
K', 'PG-13', '245000000', '2015', '393', '6.8', '2.35', '85000'], ['Color', 'Chri
stopher Nolan', '813', '164', '22000', '23000', 'Christian Bale', '27000', '44813
0642', 'Action|Thriller', 'Tom Hardy', 'The Dark Knight Rises\u00c2\u00e0
106759', 'Joseph Gordon-Levitt', '0', 'deception|imprisonment|lawlessness|police
officer|terrorist plot', 'http://www.imdb.com/title/tt1345836/?ref=fn_tt_tt_1',
'2701', 'English', 'USA', 'PG-13', '250000000', '2012', '23000', '8.5', '2.35',
'164000'], ['', 'Doug Walker', '', '', '131', '', 'Rob Walker', '131', '', 'Docum
entary', 'Doug Walker', 'Star Wars: Episode VII - The Force Awakens\u00c2\u00e0
', '8', '143', '', '0', '', 'http://www.imdb.com/title/tt5289954/?ref=fn_tt_tt_
1', '', '', '', '', '', '12', '7.1', '', '0']]

```

**ESTADO:** Completado con éxito

Ahora, querríamos ver las posiciones de los identificadores de los campos, dados en la línea cabecera.

```
In [7]: list_of_enumerated_headers = [
        (index, item) for index, item in enumerate(full_header)
    ]
```

```
In [8]: # Test de funcionamiento
        print(list_of_enumerated_headers)
```

```
[(0, 'color'), (1, 'director_name'), (2, 'num_critic_for_reviews'), (3, 'duration'), (4, 'director_facebook_likes'), (5, 'actor_3_facebook_likes'), (6, 'actor_2_name'), (7, 'actor_1_facebook_likes'), (8, 'gross'), (9, 'genres'), (10, 'actor_1_name'), (11, 'movie_title'), (12, 'num_voted_users'), (13, 'cast_total_facebook_likes'), (14, 'actor_3_name'), (15, 'facenumber_in_poster'), (16, 'plot_keywords'), (17, 'movie_imdb_link'), (18, 'num_user_for_reviews'), (19, 'language'), (20, 'country'), (21, 'content_rating'), (22, 'budget'), (23, 'title_year'), (24, 'actor_2_facebook_likes'), (25, 'imdb_score'), (26, 'aspect_ratio'), (27, 'movie_facebook_likes')]
```

**ESTADO:** Completado con éxito

## A.2. Campos principales de una película

Los campos (columnas) del archivo son demasiados. No nos interesan todos ellos. Dada una lista con todos los campos, se pide extraer otra lista sólo con los campos con los que vamos a trabajar en los siguientes apartados: `movie_title`, `title_year`, `director_name`, `actor_1_name`, `language`, `country`, `color`, `budget`, `imdb_score` y `movie_imdb_link`.

```
In [9]: indexes = [
        full_header.index(item) for item in COLUMNS_KEEP
    ] # Obtenemos los índices de aquellas columnas que queremos mantener.

def main_data_from_item(values: list):
    """
    Filtra los datos de una lista según los índices de aquellas columnas de
    interés.

    Parameters:
    -----
    values : list
        El conjunto de valores sobre los que buscar nuestros datos de interés.

    Returns:
    -----
    list
        Devuelve aquellos datos filtrados de nuestra lista.
    """

    return [
        values[item] for item in indexes
    ]
```

In [10]: *# Test de funcionamiento*

```
print(main_data_from_item(full_header))

print()

datos_avatar_2009 = main_data_from_item(full_list_data[0])
print(datos_avatar_2009)

print()

datos_star_wars_7 = main_data_from_item(full_list_data[4])
print(datos_star_wars_7)
```

```
['movie_title', 'title_year', 'director_name', 'actor_1_name', 'language', 'count
ry', 'color', 'budget', 'imdb_score', 'movie_imdb_link']
```

```
['Avatar', '2009', 'James Cameron', 'CCH Pounder', 'English', 'USA', 'Colo
r', '237000000', '7.9', 'http://www.imdb.com/title/tt0499549/?ref_=fn_tt_tt_1']
```

```
['Star Wars: Episode VII - The Force Awakens', '', 'Doug Walke
r', 'Doug Walker', '', '', '', '7.1', 'http://www.imdb.com/title/tt5289954/?r
ef_=fn_tt_tt_1']
```

**ESTADO:** Completado con éxito

### A.3. Algunos ajustes en los campos

Observa la anomalía en el string del título de la película. También, queremos tratar algunos campos como numéricos, ya sea enteros (el año y el presupuesto) o reales (la valoración). (Algunos campos numéricos enteros están vacíos en el archivo de datos; para ellos, el valor imputado será `-1`.) Además, en las urls de las películas no necesitaremos el fragmento final, iniciado con `?ref_`.

In [11]:

```
indexes_main = [
    main_data_from_item(full_header).index(column)
    for column in ['title_year', 'budget', 'imdb_score']
] # Obtenemos aquí los índices de las columnas mencionadas arriba.
```

```
def datatypes_arranged(values: list):
    """
    Modifica los elementos de la lista de entrada.
    Sobre el primer elemento, se elimina el patrón 'Â\xa0', mientras que para
    el último elemento se elimina el patrón '?ref_'. Si en algún elemento el
    valor es '' (nulo), pondremos en dicho espacio -1.

    Parameters:
    -----
    values : list
        El conjunto de valores sobre los que modificar nuestros datos.

    Returns:
    -----
    list
        Devuelve los elementos modificados en una lista, junto a los no
        modificados, en el mismo orden.
```



```

"""
values[0] = values[0][:values[0].find('Â\xa0')]

values = [
    -1 if (i in indexes_main and item == '') else values[i]
    for i, item in enumerate(values)
]

values[-1] = values[-1][:values[-1].find('?ref_')]

return values

```

In [12]: *# Test de funcionamiento*

```

print(datatypes_arranged(datos_avatar_2009))

print(datatypes_arranged(datos_star_wars_7))

```

```

['Avatar', '2009', 'James Cameron', 'CCH Pounder', 'English', 'USA', 'Color', '23
7000000', '7.9', 'http://www.imdb.com/title/tt0499549/']
['Star Wars: Episode VII - The Force Awakens', -1, 'Doug Walker', 'Doug Walker',
'', '', '', -1, '7.1', 'http://www.imdb.com/title/tt5289954/']

```

**ESTADO:** Completado con éxito

## A.4. Recuperación de alguna información básica

Diseña funciones para recuperar la siguiente información:

- El conjunto de valores posibles del campo `Color`.
- Los títulos de película de nuestro archivo (limpios de caracteres extraños), junto con la cantidad de calificadores (un entero), que contengan la subcadena "Victor".

In [26]: **def** coloresFunc(values: list):

```

"""
Se toman los valores de una lista y se devuelve un set con los
primeros elementos de una lista

Parameters:
-----
values : list
    Lista de listas o tuplas, donde se extrae el primer elemento
    de cada sublista.

Returns:
-----
set
    Conjunto de valores únicos extraídos del primer elemento de cada
    sublista de la lista de entrada.
"""
return(
    set([row[0] for row in values])
)

colores = coloresFunc(full_list_data)

```



```
def pelis_vectorFunc(values: list, header: list):
    """
    La función toma una lista de registros de películas y una lista de
    encabezados. Busca los índices de las columnas de
    'num_critic_for_reviews' y 'movie_title' en el encabezado, y
    extrae estos valores para cada película cuyo título contenga la
    palabra "Victor". El título de la película se limpia eliminando la
    cadena 'Â\x0' al final.

    Parameters:
    -----
    values : list
        Lista de listas donde cada sublista representa los datos
        de una película.

    header : list
        Lista de nombres de las columnas correspondientes a cada
        índice.

    Returns:
    -----
    list
        Lista de tuplas, cada una con el número de críticas y el título de
        la película que contiene "Victor" en su nombre.
    """

    indexes = [header.index(item)
                for item in ['num_critic_for_reviews', 'movie_title']]
    # Nos quedaremos con los índices de las entradas anteriores.

    return(
        [
            (record[indexes[0]], record[indexes[1]].replace('Â\x0', ''))
            for record in values
            if "Victor" in record[indexes[1]]
        ]
    ) # Con .replace(...) se limpian los títulos de caracteres extraños.

pelis_vector = pelis_vectorFunc(full_list_data, full_header)
```

In [27]: # Test de funcionamiento

```
print(colores)
print(pelis_vector)
```

```
{'', 'Color', ' Black and White'}
[('159', 'Victor Frankenstein'), ('159', 'Victor Frankenstein'), ('188', 'The You
ng Victoria'), ('159', 'Victor Frankenstein'), ('59', 'Raising Victor Vargas')]
```

**ESTADO:** Completado con éxito

## B Datos en un diccionario [2 puntos]

### B.1. Carga únicamente de datos que vamos a usar

En lugar de cargar *todos* los datos del archivo, luego generar otro archivo con los datos que nos interesan y luego ajustarlos, es mejor diseñar una función que recorre el archivo de datos y, de cada línea del archivo, *únicamente* retiene la parte que nos interesa, cargando la cabecera por un lado y, por otro, la lista de datos de interés, ajustados como ya hemos dicho.

Cargaremos esta información en un diccionario, cuyas claves serán pares (título, año) y cuyos valores se recogen en una lista con el resto de los campos.

Además, los datos de cada película tras la clave (título, año) están repetidas en el archivo de datos, con todos los datos idénticos; y lo que es peor: alguna que está repetida, pero con algún dato distinto. En ambos casos, para simplificar, adoptamos el criterio de registrar únicamente la primera aparición de los datos de una misma película y año.

```
In [68]: def load_main_data(url: str):
        """
        Lee un archivo externo a partir de una URL como entrada, se abre dicho
        archivo y se extrae por una parte la cabecera y por otra los datos en un
        diccionario.

        Parameters:
        -----
        url : str
            Ruta o URL del archivo CSV que contiene los datos a procesar

        Returns:
        -----
        tuple
            Un tuple que contiene dos elementos:
            - list: La cabecera seleccionada del archivo CSV.
            - dict: Un diccionario con las películas. Las claves son tuplas
              (título de la película, año de publicación) y los valores son
              listas de los datos restantes para cada película.
        """
        with open(url, encoding = 'Latin1') as file:
            # Hay que convertirlo a una lista para poder extraer los datos.
            csv_reader = list(csv.reader(file))

            header = [
                csv_reader[0][i] for i in indexes
            ]

            data = [
                datatypes_arranged(main_data_from_item(row))
                for row in csv_reader[1:]
            ] # [1:] para así no incluir la cabecera.

            movie_title_index, title_year_index = (
                header.index(column) for column in ['movie_title', 'title_year']
            )

            data_select = {
                (row[movie_title_index], row[title_year_index]): row[2:]
                for row in data
            } # [2:] porque eliminamos el nombre y el año de publicación.
```

```
return header, data_select
```

In [69]: *# Test de funcionamiento*

```
main_header, main_dict_data = load_main_data(MOVIES_DATA)

print(main_header)
print()
for title_year, pieces in list(main_dict_data.items())[:5]:
    print(title_year, " -> ", pieces)
```

```
['movie_title', 'title_year', 'director_name', 'actor_1_name', 'language', 'count
ry', 'color', 'budget', 'imdb_score', 'movie_imdb_link']
```

```
('Avatar', '2009') -> ['James Cameron', 'CCH Pounder', 'English', 'USA', 'Colo
r', '237000000', '7.9', 'http://www.imdb.com/title/tt0499549/']
('Pirates of the Caribbean: At World's End', '2007') -> ['Gore Verbinski', 'Joh
nny Depp', 'English', 'USA', 'Color', '300000000', '7.1', 'http://www.imdb.com/ti
tle/tt0449088/']
('Spectre', '2015') -> ['Sam Mendes', 'Christoph Waltz', 'English', 'UK', 'Colo
r', '245000000', '6.8', 'http://www.imdb.com/title/tt2379713/']
('The Dark Knight Rises', '2012') -> ['Christopher Nolan', 'Tom Hardy', 'Englis
h', 'USA', 'Color', '250000000', '8.5', 'http://www.imdb.com/title/tt1345836/']
('Star Wars: Episode VII - The Force Awakens', -1) -> ['Doug Walker', 'Doug Wal
ker', '', '', '', -1, '7.1', 'http://www.imdb.com/title/tt5289954/']
```

**ESTADO:** Completado con éxito

Y ahora podemos suprimir la variable `full_list_data`

In [71]: `del full_list_data`

In [72]: *# Test de funcionamiento*

```
try:
    print(full_list_data)
except:
    print('La variable full_list_data está suprimida correctamente')
```

La variable `full_list_data` está suprimida correctamente

**ESTADO:** Completado con éxito

## B.2. Recuperación de alguna información

Diseña funciones para averiguar la siguiente información:

- ¿Qué títulos de películas han sido dirigidas por "James Cameron" (o por el director que se desee)?

In [73]: `def movies_anno_for_director(dataset: dict, name: str):`

```
    """
```

```
    Filtra y devuelve las películas de un director específico, basándose
    en los datos del conjunto de películas proporcionado.
```

La función recorre un diccionario en el que las claves son identificadores o títulos de películas y los valores son listas de detalles asociados a cada película. Si el nombre del director coincide con el nombre especificado la película es añadida a una lista de películas seleccionadas.

Parameters:

-----

dataset : dict

Un diccionario donde las claves representan películas y los valores son listas de datos asociados, con el nombre del director como el primer elemento.

name : str

El nombre del director cuyas películas se desean seleccionar.

Returns:

-----

list

Una lista de claves del diccionario (títulos o identificadores de películas) correspondientes a las películas dirigidas por el director especificado.

"""

```
movies_selected = [
    row for row, values in zip(dataset, dataset.values())
    if values[0] == name
]
```

```
return movies_selected
```

In [74]: *# Test de funcionamiento*

```
movies_anno_for_director(main_dict_data, "James Cameron")
```

```
Out[74]: [('Avatar', '2009'),
          ('Titanic', '1997'),
          ('Terminator 2: Judgment Day', '1991'),
          ('True Lies', '1994'),
          ('The Abyss', '1989'),
          ('Aliens', '1986'),
          ('The Terminator', '1984')]
```

**ESTADO:** Completado con éxito

- ¿Qué directores han dirigido el número máximo de películas?

In [75]: **def** directors\_max\_movies(dataset: dict):

"""

Encuentra el director con el número máximo de películas dirigidas.

Parameters:

-----

dataset : dict

Un diccionario donde cada valor contiene una lista de los detalles de la película, con el nombre del director como primer elemento.

Returns:

```

-----
tuple
    Una tupla conteniendo una lista con el nombre del director
    que ha dirigido el número máximo de películas.
"""
# Obtenemos la lista de todos los actores que no sean nulos.
all_names = [row[0] for row in dataset.values() if row[0]]

# Contamos la frecuencia de cada actor.
director_counts = Counter(all_names)

# Hallamos el número máximo de películas dirigidas.
max_value = max(director_counts.values())

# Obtenemos el director con el número máximo de películas dirigidas.
max_directors = [
    director for director, count in director_counts.items()
    if count == max_value
]

return max_directors, max_value

```

```

In [76]: # Test de funcionamiento

print(directors_max_movies(main_dict_data))

```

(['Steven Spielberg'], 26)

### ESTADO: Completado con éxito

- Para cada año de un intervalo dado de años, ¿cuántas películas se han realizado? Esta información debe recuperarse en un diccionario convencional (no por defecto), y luego debe mostrarse en una lista con los años en orden ascendente..

```

In [77]: def years_num_movies(dataset: dict, min_year: int, max_year: int):
    """
    Cuenta el número de películas producidas por año dentro de un rango
    específico.

    Recorre el conjunto de datos y cuenta cuántas películas fueron producidas
    en cada año dentro del rango especificado por `min_year` y `max_year`.
    El resultado es un diccionario donde las claves son los años y los valores
    son la cantidad de películas para ese año.

    Parameters:
    -----
    dataset : dict
        Diccionario con las películas, donde cada clave es una tupla que
        contiene el título de la película y el año de producción.
    min_year : int
        El año mínimo del rango a considerar.
    max_year : int
        El año máximo del rango a considerar.

    Returns:
    -----
    dict
    """

```

```

        Un diccionario donde las claves son los años y los valores son la
        cantidad de películas producidas en cada año dentro del rango.
        """
        return {
            year: sum(1 for key in dataset.keys() if key[1] == f'{year}')
            for year in range(min_year, max_year + 1)
        }

```

In [78]: *# Test de funcionamiento*

```

num_movies = years_num_movies(main_dict_data, 2000, 2015)

print(num_movies)

```

```

{2000: 169, 2001: 183, 2002: 204, 2003: 169, 2004: 207, 2005: 216, 2006: 235, 2007: 198, 2008: 223, 2009: 253, 2010: 225, 2011: 224, 2012: 214, 2013: 231, 2014: 243, 2015: 211}

```

**ESTADO:** Completado con éxito

In [79]: *# Ahora solo queda ajustar el diccionario ordenado al formato pedido.*

```

num_movies_sorted = [
    (year, num_movies[year]) for year in list(num_movies.keys())
]

```

In [80]: *# Test de funcionamiento*

```

print(num_movies_sorted)

```

```

[(2000, 169), (2001, 183), (2002, 204), (2003, 169), (2004, 207), (2005, 216), (2006, 235), (2007, 198), (2008, 223), (2009, 253), (2010, 225), (2011, 224), (2012, 214), (2013, 231), (2014, 243), (2015, 211)]

```

**ESTADO:** Completado con éxito

- Diseña también una operación que, partiendo de nuestro diccionario de los datos principales, almacene en un archivo los datos siguientes de cada película, sin la cabecera: el título, el idioma, el año, el país y el presupuesto. El separador será en este caso el carácter `|`. El archivo usado para el almacenamiento es el siguiente:

```
FEW_FIELDS = "algunos_campos.txt"
```

In [81]: **def** store\_file(dataset: dict, output\_file\_name: str):

```

    """
    Almacena los datos de un conjunto de películas en un archivo de texto.

    Escribe los detalles seleccionados de cada película en un archivo de salida.
    Cada fila del archivo contiene el título de la película, el año de
    lanzamiento, junto con algunos otros valores del dataset,
    separados por el carácter "|".

    Parameters:
    -----
    dataset : dict
        Un diccionario donde las claves son tuplas que contienen el título de
        la película y el año de lanzamiento, y los valores son listas con

```

```

    detalles adicionales de la película.
    output_file_name : str
        El nombre del archivo de salida donde se almacenarán los datos.

    Returns:
    -----
    None
    """
    with open(output_file_name, 'w', encoding = 'Latin1') as write_file:
        for key, values in zip(dataset, dataset.values()):
            write_file.write(
                f"{key[0]}|{key[1]}|{values[2]}|{values[3]}|{values[5]}\n"
            )

```

```

In [82]: # Test de funciomaniemto

store_file(main_dict_data, FEW_FIELDS)

! dir algunos*.*

print()

with open(FEW_FIELDS) as f:
    for i in range(5):
        print(f.readline())

```

El volumen de la unidad C es Windows  
El número de serie del volumen es: CA8A-947A

Directorio de C:\Users\Usuario

```

23/10/2024  11:25          210.862 algunos_campos.txt
              1 archivos          210.862 bytes
              0 dirs  23.938.805.760 bytes libres

```

Avatar|2009|English|USA|237000000

Pirates of the Caribbean: At World's End|2007|English|USA|300000000

Spectre|2015|English|UK|245000000

The Dark Knight Rises|2012|English|USA|250000000

Star Wars: Episode VII - The Force Awakens|-1|||-1

**ESTADO:** Completado con éxito

### B.3. Un conteo sencillo con `defaultdict`

Deseamos saber con qué directores y número de veces ha actuado cada actor como actor principal. Se pide realizar este conteo en un `defaultdict` cuyas claves serán nombres de los actores y cuyos valores tendrán la estructura de un `defaultdict`,



cuyas claves serán los nombres de los directores y cuyos valores serán los números contabilizados.

```
In [83]: def actor_directors(dataset: dict):
        """
        Creamos un diccionario que contenga como clave el nombre de un actor
        y como valores otro diccionario que contenga el nombre del director
        con el que ha trabajado como clave y el número de veces como valores.

        Parameters:
        -----
        dataset : dict
            Un diccionario donde cada valor contiene detalles de la película,
            donde el primer elemento es el nombre del director y el segundo el
            nombre del actor

        Returns:
        -----
        dict
            Un diccionario que tenga como clave el nombre del actor y como valores
            un diccionario con los nombres de los directores con los que ha
            trabajado como clave y el número de veces como valores.
        """
        dict_total = defaultdict(lambda: defaultdict(int))

        for row in dataset.values():
            director_name = row[0]
            actor_name = row[1]

            if actor_name: # Contamos solo para aquellas entradas no nulas.
                dict_total[actor_name][director_name] += 1

        return dict_total
```

```
In [84]: # Test de funcionamiento

num_collaborations = actor_directors(main_dict_data)

print(type(num_collaborations))
key_a, value_a = list(num_collaborations.items())[0]
print(type(key_a), type(value_a))
key_b, value_b = list(value_a.items())[0]
print(type(key_b), type(value_b))

print()

print(num_collaborations)
```

```
<class 'collections.defaultdict'>  
<class 'str'> <class 'collections.defaultdict'>  
<class 'str'> <class 'int'>  
  
defaultdict(<function actor_directors.<locals>.<lambda> at 0x00000210C45CCCC0>, {  
'CCH Pounder': defaultdict(<class 'int'>, {'James Cameron': 1, 'Peter Hyams': 1,  
'Fred Dekker': 1, 'Ernest R. Dickerson': 1}), 'Johnny Depp': defaultdict(<class  
'int'>, {'Gore Verbinski': 5, 'Rob Marshall': 2, 'Tim Burton': 6, 'James Bobin':  
1, 'Michael Mann': 1, 'Florian Henckel von Donnersmarck': 1, 'Wally Pfister': 1,  
'David Koepp': 2, 'Scott Cooper': 1, 'Albert Hughes': 1, 'Roman Polanski': 1, 'We  
s Craven': 1, 'Mike Newell': 1, 'Rand Ravich': 1, 'Ted Demme': 1, 'Robert Rodrigu  
ez': 1, 'Marc Forster': 1, 'Jeremy Leven': 1, 'Laurence Dunmore': 1, 'Terry Gilli  
am': 1, 'Lasse Hallstr m': 1, 'Oliver Stone': 1, 'Rachel Talalay': 1, 'Kevin Smi  
th': 2}), 'Christoph Waltz': defaultdict(<class 'int'>, {'Sam Mendes': 1, 'David  
Yates': 1, 'Michel Gondry': 1, 'Tim Burton': 1}), 'Tom Hardy': defaultdict(<class  
'int'>, {'Christopher Nolan': 1, 'George Miller': 1, 'McG': 1, 'Stuart Baird': 1,  
'Daniel Espinosa': 1, 'Brian Helgeland': 1, "Gavin O'Connor": 1, 'Guy Ritchie':  
1, '': 1, 'Matthew Vaughn': 1, 'Nicolas Winding Refn': 1}), 'Doug Walker': defaul  
tdict(<class 'int'>, {'Doug Walker': 1}), 'Daryl Sabara': defaultdict(<class 'in  
t'>, {'Andrew Stanton': 1, 'Eli Roth': 1}), 'J.K. Simmons': defaultdict(<class 'i  
nt'>, {'Sam Raimi': 4, 'Alan Taylor': 1, 'Alessandro Carloni': 1, 'Tom McGrath':  
1, 'Joe Johnston': 1, 'Michael Caton-Jones': 1, 'John Hamburg': 1, 'Gore Verbinsk  
i': 1, 'Joan Chen': 1, 'Michael Apted': 1, 'Ethan Coen': 2, 'Jason Reitman': 4,  
'Gavin Hood': 1, 'Baltasar Korm kur': 1, 'Karyn Kusama': 1, 'Vicky Jenson': 1,  
'Michael Polish': 1, 'Jonas Elmer': 1, 'Mike Judge': 1, 'Brian Klugman': 1, 'Dami  
en Chazelle': 1, 'David Ayer': 1}), 'Brad Garrett': defaultdict(<class 'int'>,  
{'Nathan Greno': 1, 'Marc Lawrence': 1}), 'Chris Hemsworth': defaultdict(<class  
'int'>, {'Joss Whedon': 2, 'Rupert Sanders': 1, 'Alan Taylor': 1, 'Kenneth Branag  
h': 1, 'J.J. Abrams': 1, 'Cedric Nicolas-Troyan': 1, 'Ron Howard': 2, 'Michael Ma  
nn': 1, 'John Francis Daley': 1, 'Drew Goddard': 1, 'David Twohy': 1, 'Stephen Mi  
lburn Anderson': 1}), 'Alan Rickman': defaultdict(<class 'int'>, {'David Yates':  
1, 'Kevin Reynolds': 1, 'Dean Parisot': 1, 'John McTiernan': 1, 'Neil Jordan': 1,  
'Simon Wincer': 1, 'Ang Lee': 1, 'Randall Miller': 1}), 'Henry Cavill': defaultdi  
ct(<class 'int'>, {'Zack Snyder': 2, 'Guy Ritchie': 1, 'Tarsem Singh': 1, 'Matthe  
w Vaughn': 1, 'Kevin Reynolds': 1, 'Mabrouk El Mechri': 1}), 'Kevin Spacey': defa  
ultdict(<class 'int'>, {'Bryan Singer': 2, 'Wolfgang Petersen': 1, 'F. Gary Gra  
y': 1, 'Alan Parker': 1, 'Iain Softley': 1, 'John Lasseter': 1, 'Joel Schumache  
r': 1, 'Sean Anders': 1, 'Mimi Leder': 1, 'Seth Gordon': 1, 'Robert Luketic': 1,  
'Curtis Hanson': 1, 'Clint Eastwood': 1, 'Lasse Hallstr m': 1, 'Grant Heslov':  
1, 'Kevin Spacey': 1, 'Sam Mendes': 1, 'James Foley': 1, 'George Hickenlooper':  
1, 'Duncan Jones': 1, 'J.C. Chandor': 1}), 'Giancarlo Giannini': defaultdict(<cla  
ss 'int'>, {'Marc Forster': 1}), 'Peter Dinklage': defaultdict(<class 'int'>, {'A  
ndrew Adamson': 1, 'Steve Martino': 1, 'Chris Columbus': 1, 'Clay Kaytis': 1, 'Jo  
n Favreau': 1, 'Ben Falcone': 1, 'Frank Oz': 1, 'Sidney Lumet': 1, 'Hue Rhodes':  
1, 'Joseph Kosinski': 1, 'Tom McCarthy': 1, 'Eric Schaeffer': 1}), 'Will Smith':  
defaultdict(<class 'int'>, {'Barry Sonnenfeld': 4, 'David Ayer': 1, 'Peter Berg':  
1, 'Francis Lawrence': 1, 'Michael Bay': 2, 'M. Night Shyamalan': 1, 'Michael Man  
n': 1, 'Alex Proyas': 1, 'Tony Scott': 1, 'Roland Emmerich': 1, 'Glenn Ficarra':  
1, 'Andy Tennant': 1, 'Gabriele Muccino': 2, 'Peter Landesman': 1}), 'Aidan Turne  
r': defaultdict(<class 'int'>, {'Peter Jackson': 3, 'Harald Zwart': 1}), 'Emma St  
one': defaultdict(<class 'int'>, {'Marc Webb': 2, 'Tom Dey': 1, 'Cameron Crowe':  
1, 'Tate Taylor': 1, 'Fred Wolf': 1, 'Ruben Fleischer': 1, 'Alejandro G. I   rritu':  
1, 'Greg Mottola': 1, 'Peter Cattaneo': 1, 'Will Gluck': 1}), 'Mark Addy': d  
efaultdict(<class 'int'>, {'Ridley Scott': 1, 'Simon Wells': 1, 'Richard J. Lewi  
s': 1}), 'Christopher Lee': defaultdict(<class 'int'>, {'Chris Weitz': 1, 'Peter  
Jackson': 2, 'Joe Dante': 1, 'Steven Spielberg': 1, 'James Fargo': 1, 'Alan Mette  
r': 1, 'Guy Hamilton': 1, 'Harald Reinl': 1}), 'Naomi Watts': defaultdict(<class  
'int'>, {'Peter Jackson': 1, 'Robert Schwentke': 1, 'Hideo Nakata': 1, 'Tom Tykw  
e': 1, 'Jim Sheridan': 1, 'Doug Liman': 1, 'J.A. Bayona': 1, 'Rachel Talalay': 1,  
'David O. Russell': 1, 'Alejandro G. I   rritu': 1, 'David Lynch': 1, 'Michael H
```

```

aneke': 1, 'Noah Baumbach': 1, 'Rodrigo Garc  \xada': 1}}, 'Leonardo DiCaprio': de
faultdict(<class 'int'>, {'James Cameron': 1, 'Baz Luhrmann': 2, 'Christopher Nol
an': 1, 'Alejandro G. I    rritu': 1, 'Martin Scorsese': 5, 'Quentin Tarantino':
1, 'Edward Zwick': 1, 'Ridley Scott': 1, 'Steven Spielberg': 1, 'Danny Boyle': 1,
'Sam Mendes': 1, 'Randall Wallace': 1, 'Clint Eastwood': 1, 'Sam Raimi': 1, 'Jerr
y Zaks': 1}}, 'Robert Downey Jr.': defaultdict(<class 'int'>, {'Anthony Russo':
1, 'Shane Black': 2, 'Jon Favreau': 2, 'Guy Ritchie': 2, 'Ben Stiller': 1, 'David
Fincher': 1, 'Todd Phillips': 1, 'Brian Robbins': 1, 'Joe Wright': 1, 'Frank Oz':
1, 'Curtis Hanson': 2, 'David Dobkin': 1, 'Mathieu Kassovitz': 1, 'Neil Jordan':
1, 'Jodie Foster': 1, 'Richard Linklater': 1, 'Steven Shainberg': 1, 'Jon Poll':
1, 'Richard Loncraine': 1, 'Keith Gordon': 1, 'George Clooney': 1, 'James Tobac
k': 1}}, 'Liam Neeson': defaultdict(<class 'int'>, {'Peter Berg': 1, 'Jonathan Li
ebesman': 1, 'Louis Leterrier': 1, 'Ridley Scott': 1, 'Kathryn Bigelow': 1, 'Jan
de Bont': 1, 'Seth MacFarlane': 2, 'Jaume Collet-Serra': 3, 'Olivier Megaton': 2,
'Peter Yates': 1, 'Peter Lepeniotis': 1, 'Michael Caton-Jones': 1, 'Scott Frank':
1, 'Pierre Morel': 1, 'Steven Spielberg': 1, 'Joe Carnahan': 1, 'Roger Donaldso
n': 1, 'Anthony Silverston': 1, 'Atom Egoyan': 1, 'John H. Lee': 1, 'Bill Condo
n': 1, 'Eric Blakeney': 1, 'Agnieszka Wojtowicz-Vosloo': 1}}, 'Bryce Dallas Howar
d': defaultdict(<class 'int'>, {'Colin Trevorrow': 1, 'M. Night Shyamalan': 2, 'L
ars von Trier': 1, 'David Lowery': 1, 'Marianna Palka': 1}}, 'Albert Finney': def
aultdict(<class 'int'>, {'Sam Mendes': 1, 'Tony Richardson': 1}}, 'Hugh Jackman':
defaultdict(<class 'int'>, {'Brett Ratner': 1, 'Gavin Hood': 1, 'David Bowers':
1, 'Joe Wright': 1, 'Peter Ramsey': 1, 'Bryan Singer': 2, 'James Mangold': 2, 'Sh
awn Levy': 1, 'Dominic Sena': 1, 'Tom Hooper': 1, 'Neill Blomkamp': 1, 'Denis Vil
leneuve': 1, 'Darren Aronofsky': 1, 'Dexter Fletcher': 1, 'Tony Goldwyn': 1, 'Eli
zabeth Banks': 1}}, 'Steve Buscemi': defaultdict(<class 'int'>, {'Dan Scanlon':
1, 'Hironobu Sakaguchi': 1, 'Pete Docter': 1, 'Will Finn': 1, 'Genndy Tartakovsk
y': 2, 'Gary Winick': 1, 'Dennis Dugan': 3, 'Simon West': 1, 'Gil Kenan': 1, 'Tim
Burton': 1, 'Harold Becker': 1, 'Steven Brill': 1, 'John Carpenter': 1, 'Betty Th
omas': 1, 'Robert Rodriguez': 1, 'Don Scardino': 1, 'Robert Altman': 1, 'Miguel A
rteta': 1, 'Gary Fleder': 1, 'Chris Rock': 1, 'Olivier Assayas': 1, 'Oren Moverma
n': 1, 'Jonathan Wacks': 1, 'Steve Buscemi': 1}}, 'Glenn Morshower': defaultdict
(<class 'int'>, {'Michael Bay': 2, 'Jon Amiel': 1}}, 'Bingbing Li': defaultdict(<
class 'int'>, {'Michael Bay': 1, 'Li Zhang': 1, 'Wayne Wang': 1}}, 'Tim Holmes':
defaultdict(<class 'int'>, {'Sam Raimi': 1}}, 'Jeff Bridges': defaultdict(<class
'int'>, {'Joseph Kosinski': 1, 'Ash Brannon': 1, 'Sergey Bodrov': 1, 'Gary Ross':
1, 'Mark Osborne': 1, 'Michael Cimino': 1, 'Barbra Streisand': 1, 'Ridley Scott':
1, 'Phillip Noyce': 1, 'Terry Gilliam': 1, 'Mark Pellington': 1, 'Jessica Bending
er': 1, 'Albert Brooks': 1, 'Michael Meredith': 1, 'Rod Lurie': 1, 'Scott Coope
r': 1, 'Larry Charles': 1}}, 'Joe Mantegna': defaultdict(<class 'int'>, {'John La
sseter': 1, 'Jon Avnet': 1, 'Patrick Read Johnson': 1, 'John Schlesinger': 1, 'To
m Holland': 1, 'Stuart Gordon': 1, 'Charlie Levi': 1}}, 'Ryan Reynolds': defaultd
ict(<class 'int'>, {'Martin Campbell': 1, 'David Soren': 1, 'Robert Schwentke':
1, 'Kirk De Micco': 1, 'David S. Goyer': 1, 'Tim Miller': 1, 'David Dobkin': 1,
'Anne Fletcher': 1, 'Tarsem Singh': 1, 'Joe Carnahan': 1, 'Andrew Fleming': 1, 'S
imon Curtis': 1, 'Adam Brooks': 1, 'Walt Becker': 1, 'Rodrigo Cort  s': 1, 'Rob M
cKittrick': 1}}, 'Tom Hanks': defaultdict(<class 'int'>, {'Lee Unkrich': 1, 'Robe
rt Zemeckis': 3, 'Ron Howard': 4, 'Tom Tykwer': 1, 'John Lasseter': 2, 'Sam Mende
s': 1, 'Steven Spielberg': 3, 'Nora Ephron': 1, 'Frank Darabont': 1, 'Paul Greeng
rass': 1, 'Penny Marshall': 2, 'Stephen Daldry': 1, 'John Lee Hancock': 1, 'Tom H
anks': 2}}, 'Christian Bale': defaultdict(<class 'int'>, {'McG': 1, 'Christopher
Nolan': 3, 'Ridley Scott': 1, 'Yimou Zhang': 1, 'Rob Bowman': 1, 'John Madden':
1, 'Mike Gabriel': 1, 'John Singleton': 1, 'James Mangold': 1, 'Terrence Malick':
1, 'Hayao Miyazaki': 1, 'Scott Cooper': 1, 'Todd Haynes': 2, 'Kurt Wimmer': 1, 'G
illian Armstrong': 1, 'David O. Russell': 1, 'Werner Herzog': 1, 'Mary Harron':
1, 'Brad Anderson': 1}}, 'Jason Statham': defaultdict(<class 'int'>, {'James Wa
n': 1, 'Simon West': 3, 'Patrick Hughes': 1, 'Sylvester Stallone': 1, 'Gary McKen
dry': 1, 'Paul W.S. Anderson': 1, 'F. Gary Gray': 1, 'Uwe Boll': 1, 'James Wong':
1, 'Taylor Hackford': 1, 'Boaz Yakin': 1, 'Louis Leterrier': 2, 'John Carpenter':
1, 'Philip G. Atwell': 1, 'Gary Fleder': 1, 'Roger Donaldson': 1, 'Mark Neveldin

```

```

e': 2, 'Guy Ritchie': 2, 'Barry Skolnick': 1, 'Hunter Richards': 1}}, 'Peter Capa
ldi': defaultdict(<class 'int'>, {'Marc Forster': 1}), 'Jennifer Lawrence': defau
ltdict(<class 'int'>, {'Bryan Singer': 2, 'Matthew Vaughn': 1, 'Francis Lawrenc
e': 3, 'Gary Ross': 1, 'David O. Russell': 3, 'Jodie Foster': 1, 'Mark Tonderai':
1, 'Debra Granik': 1, 'Lori Petty': 1, 'Drake Doremus': 1}), 'Benedict Cumberbatc
h': defaultdict(<class 'int'>, {'J.J. Abrams': 1, 'Eric Darnell': 1, 'Joe Wrigh
t': 1, 'Bill Condon': 1, 'John Wells': 1, 'Tomas Alfredson': 1, 'Morten Tyldum':
1}), 'Eddie Marsan': defaultdict(<class 'int'>, {'Bryan Singer': 1, 'Mike Leigh':
1}), 'Jake Gyllenhaal': defaultdict(<class 'int'>, {'Mike Newell': 1, 'Roland Emm
erich': 1, 'Sam Mendes': 1, 'Duncan Jones': 1, 'Antoine Fuqua': 1, 'Edward Zwic
k': 1, 'David O. Russell': 1, 'Brad Silberling': 1, 'David Ayer': 1, 'Ang Lee':
1, 'Blair Hayes': 1, 'Dan Gilroy': 1, 'Miguel Arteta': 1, 'Richard Kelly': 1, 'Ja
mes Cox': 1}), 'Charlie Hunnam': defaultdict(<class 'int'>, {'Guillermo del Tor
o': 1, 'Alfonso CuarÃ³n': 1, 'Stephen Gaghan': 1, 'Douglas McGrath': 1, 'Stefan R
uzowitzky': 1}), 'Harrison Ford': defaultdict(<class 'int'>, {'Steven Spielberg':
4, 'Gavin Hood': 1, 'Robert Zemeckis': 1, 'Alan J. Pakula': 1, 'Wolfgang Peterse
n': 1, 'Ivan Reitman': 1, 'Ron Shelton': 1, 'Sydney Pollack': 1, 'Phillip Noyce':
2, 'Adam McKay': 1, 'Andrew Davis': 1, 'Brian Helgeland': 1, 'Richard Marquand':
1, 'Francis Ford Coppola': 2, 'Tom Vaughan': 1, 'Lee Toland Krieger': 1, 'Ridley
Scott': 1, 'Irvin Kershner': 1, 'Peter Weir': 1, 'George Lucas': 2, 'Guy Hamilto
n': 1}), 'A.J. Buckley': defaultdict(<class 'int'>, {'Peter Sohn': 1}), 'Kelly Ma
cdonald': defaultdict(<class 'int'>, {'Mark Andrews': 1, 'Joe Wright': 1, 'Ethan
Coen': 1, 'Mike Figgis': 1, 'Danny Boyle': 1}), 'Sofia Boutella': defaultdict(<cl
ass 'int'>, {'Justin Lin': 1}), 'John Ratzenberger': defaultdict(<class 'int'>,
{'Andrew Stanton': 1, 'Pete Docter': 1, 'John Lasseter': 1}), 'Tzi Ma': defaultdi
ct(<class 'int'>, {'Brett Ratner': 1}), 'Oliver Platt': defaultdict(<class 'in
t'>, {'Roland Emmerich': 1, 'Betty Thomas': 1, 'Stuart Baird': 1, 'Gary Fleder':
1, 'Steve Miner': 1, 'Menno Meyjes': 1, 'Brian Robbins': 1, 'Mark Steven Johnso
n': 1, 'Nicole Holofcener': 1, 'Peter Hedges': 1}), 'Robin Wright': defaultdict(<
class 'int'>, {'Robert Zemeckis': 2, 'David Fincher': 1, 'M. Night Shyamalan': 1,
'Kevin Macdonald': 1, 'Luis Mandoki': 1, 'Robert Redford': 1, 'Rob Reiner': 1, 'M
ichael Mayer': 1, 'Anne Fontaine': 1}), 'Channing Tatum': defaultdict(<class 'in
t'>, {'Lana Wachowski': 1, 'Roland Emmerich': 1, 'Jon M. Chu': 1, 'Ron Howard':
1, 'Phil Lord': 2, 'Jorge R. GutiÃ©rrez': 1, 'Thomas Carter': 1, 'Evan Goldberg':
1, 'Michael Sucsy': 1, 'Steven Soderbergh': 3, 'Steve Boyum': 1, 'Andy Fickman':
1, 'Lasse HallstrÃ¶m': 1, 'Dito Montiel': 1, 'Anne Fletcher': 1}), 'Jim Broadben
t': defaultdict(<class 'int'>, {'Andrew Adamson': 1, 'Frank Coraci': 1, 'Sarah Sm
ith': 1, 'Chris Wedge': 1, 'Baz Luhrmann': 1, 'Gary Chapman': 1, 'Bille August':
1, 'Reinhard Klooss': 1, 'Peter Hewitt': 1, 'Woody Allen': 1, 'Mike Leigh': 2, 'S
idney J. Furie': 1, 'Ian Fitzgibbon': 1, '': 1, 'Mark Herman': 1, 'Neil Jordan':
1}), 'Amy Poehler': defaultdict(<class 'int'>, {'Rob Letterman': 1, 'Pete Docte
r': 1, 'Mike Mitchell': 2, 'Betty Thomas': 1, 'Jimmy Hayward': 1, 'Mike Disa': 1,
'Barry Levinson': 1}), 'ChloÃ© Grace Moretz': defaultdict(<class 'int'>, {'Martin
Scorsese': 1, 'Byron Howard': 1, 'John Whitesell': 1, 'J Blakeson': 1, 'Nicholas
Stoller': 1, 'Kimberly Peirce': 1, 'Jeff Wadlow': 1, 'Matt Reeves': 1, 'Thor Freu
denthal': 1, 'David Moreau': 1, 'R.J. Cutler': 1}), 'Jet Li': defaultdict(<class
'int'>, {'Rob Cohen': 1, 'Richard Donner': 1, 'Rob Minkoff': 1, 'Yimou Zhang': 1,
'Peter Ho-Sun Chan': 1, 'Andrzej Bartkowiak': 2, 'Chris Nahon': 1}), 'Jimmy Benne
tt': defaultdict(<class 'int'>, {'Tom Shadyac': 1, 'Wolfgang Petersen': 1, 'Flore
nt-Emilio Siri': 1, 'Richard Loncraine': 1, 'Jaume Collet-Serra': 1, 'Robert Rodr
iguez': 1, 'Andrew Douglas': 1, 'James Mottern': 1}), 'Tom Cruise': defaultdict(<
class 'int'>, {'Doug Liman': 1, 'Christopher McQuarrie': 2, 'Brad Bird': 1, 'Edwa
rd Zwick': 1, 'Steven Spielberg': 2, 'John Woo': 1, 'James Mangold': 1, 'Bryan Si
nger': 1, 'Brian De Palma': 1, 'Cameron Crowe': 2, 'Stanley Kubrick': 1, 'Tony Sc
ott': 2, 'Michael Mann': 1, 'Sydney Pollack': 1, 'Rob Reiner': 1, 'Barry Levinso
n': 1, 'Oliver Stone': 1, 'Martin Scorsese': 1, 'Francis Ford Coppola': 1, 'Curti
s Hanson': 1}), 'Jeanne Tripplehorn': defaultdict(<class 'int'>, {'Kevin Reynold
s': 1, 'Kelly Makin': 1, 'Peter Howitt': 1}), 'Joseph Gordon-Levitt': defaultdict
(<class 'int'>, {'Stephen Sommers': 1, 'Ron Clements': 1, 'Steven Spielberg': 1,
'Frank Miller': 1, 'Spike Lee': 1, 'Brian Gibson': 1, '': 1, 'David Koepp': 1, 'R

```

```

obert Zemeckis': 1, 'Rian Johnson': 1, 'Kimberly Peirce': 1, 'Gil Junger': 1, 'Jo
nathan Levine': 1, 'Marc Webb': 1, 'Spencer Susser': 1, 'Joseph Gordon-Levitt':
1, 'C. Jay Cox': 1}}, 'Scarlett Johansson': defaultdict(<class 'int'>, {'Jon Favr
eau': 1, 'Anthony Russo': 1, 'Michael Bay': 1, 'Robert Redford': 1, 'Brian De Pal
ma': 1, 'Frank Miller': 1, 'Cameron Crowe': 1, 'Luc Besson': 1, 'Stephen Hillenbu
rg': 1, 'Ellory Elkayem': 1, 'Paul Weitz': 1, 'Spike Jonze': 1, 'Ethan Coen': 1,
'Woody Allen': 3, 'Jonathan Glazer': 1, 'Peter Webber': 1, 'Terry Zwigoff': 1, 'S
ofia Coppola': 1}}, 'Angelina Jolie Pitt': defaultdict(<class 'int'>, {'Robert St
romberg': 1, 'Jennifer Yuh Nelson': 1, 'Mark Osborne': 1, 'Phillip Noyce': 1, 'Si
mon West': 1, 'Timur Bekmambetov': 1, 'Kerry Conran': 1, 'Martin Campbell': 1, 'C
lint Eastwood': 1, 'Stephen Herek': 1, 'Iain Softley': 1, 'Mike Newell': 1, 'Mich
ael Cristofer': 1, 'James Mangold': 1, 'Michael Winterbottom': 1}}, 'Gary Oldma
n': defaultdict(<class 'int'>, {'Matt Reeves': 1, 'JosÃ© Padilha': 1, 'Stephen Ho
pkins': 1, 'Roland JoffÃ©': 1, 'Catherine Hardwicke': 1, 'Frederik Du Chau': 1,
'Ariel Vromen': 1, 'David S. Goyer': 1, 'Peter Medak': 1, 'Julian Schnabel': 1}},
'Tamsin Egerton': defaultdict(<class 'int'>, {'Roland JoffÃ©': 1}}, 'Keanu Reeve
s': defaultdict(<class 'int'>, {'Carl Rinsch': 1, 'Nancy Meyers': 1, 'Scott Derri
ckson': 1, 'Lana Wachowski': 1, 'Taylor Hackford': 1, 'Andrew Davis': 1, 'Howard
Deutch': 1, 'Francis Ford Coppola': 1, 'Alejandro Agresti': 1, "Pat O'Connor": 1,
'Joe Charbanic': 1, 'Jan de Bont': 1, 'Stephen Frears': 1, 'Brian Robbins': 1, 'P
eter Hewitt': 1, 'David Ayer': 1, 'Stephen Herek': 1, 'Kenneth Branagh': 1, 'Nico
las Winding Refn': 1, 'Stephen Kay': 1, 'Gus Van Sant': 1, 'Tim Hunter': 1}}, 'Jo
n Hamm': defaultdict(<class 'int'>, {'Mike Mitchell': 1, 'Zack Snyder': 1, 'Randa
ll Wallace': 1, 'Craig Gillespie': 1, 'David Wain': 1}}, 'Judy Greer': defaultdic
t(<class 'int'>, {'Brad Bird': 1, 'Peyton Reed': 1, 'Nancy Meyers': 1, 'Mike Nich
ols': 1, 'David O. Russell': 1, 'Anne Fletcher': 1, 'Brandon Camp': 1, 'Paul Weit
z': 1, 'Jay Duplass': 1, "Chris D'Arienzo": 1, 'Adam Goldberg': 1, 'Jonathan Kess
elman': 1, 'Craig Mazin': 1}}, 'Damon Wayans Jr.': defaultdict(<class 'int'>, {'D
on Hall': 1, 'Damien Dante Wayans': 1}}, 'Jack McBrayer': defaultdict(<class 'in
t'>, {'Rich Moore': 1, 'Brad Peyton': 1, 'Ramaa Mosley': 1}}, 'Vivica A. Fox': de
faultdict(<class 'int'>, {'Roland Emmerich': 1, 'Quentin Tarantino': 1, 'Jesse Va
ughan': 1, 'Mort Nathan': 1, 'Rodman Flender': 1, 'F. Gary Gray': 1, 'Mark Brow
n': 1, 'David Boyd': 1}}, 'Gerard Butler': defaultdict(<class 'int'>, {'Dean DeBl
ois': 2, 'Alex Proyas': 1, 'Jan de Bont': 1, 'Antoine Fuqua': 1, 'Zack Snyder':
1, 'Babak Najafi': 1, 'Joel Schumacher': 1, 'F. Gary Gray': 1, 'Mark Neveldine':
1, 'Andy Tennant': 1, 'Robert Luketic': 1, 'Jennifer Flackett': 1, 'Gabriele Mucc
ino': 1, 'Richard LaGravenese': 1, 'Marc Forster': 1, 'Patrick Lussier': 1, 'Mich
ael Apted': 1, 'David Anspaugh': 1, 'Mike Barker': 1, 'Ralph Fiennes': 1, 'Shona
Auerbach': 1}}, 'Nick Stahl': defaultdict(<class 'int'>, {'Jonathan Mostow': 1,
'Terrence Malick': 1}}, 'Bradley Cooper': defaultdict(<class 'int'>, {'James Gun
n': 1, 'Todd Phillips': 2, 'Clint Eastwood': 1, 'Garry Marshall': 1, 'Peyton Ree
d': 1, 'Tom Dey': 1, 'Christian Alvart': 1, 'David Palmer': 1, 'John Wells': 1,
'Phil Traill': 1, 'RyÅ»hei Kitamura': 1, 'Dan Trachtenberg': 1}}, 'Matthew McCona
ughey': defaultdict(<class 'int'>, {'Christopher Nolan': 1, 'Breck Eisner': 1, 'R
obert Zemeckis': 1, 'Andy Tennant': 1, 'McG': 1, 'Jonathan Mostow': 1, 'Ron Howar
d': 1, 'Donald Petrie': 1, 'Gary Ross': 1, 'Brad Furman': 1, 'Adam Shankman': 1,
'Richard Linklater': 2, 'Bill Paxton': 1, 'Jeff Nichols': 1, 'William Friedkin':
1, 'S.R. Bindler': 1, 'Jean-Marc VallÃ©': 1, 'John Sayles': 1, 'Jill Sprecher':
1}}, 'Mark Chinnery': defaultdict(<class 'int'>, {'Hideaki Anno': 1}}, 'Paul Walk
er': defaultdict(<class 'int'>, {'Rob Cohen': 2, 'Justin Lin': 1, 'Richard Donne
r': 1, 'John Singleton': 1, 'Clint Eastwood': 1, 'John Stockwell': 1, 'Frank Mars
hall': 1, 'John Luessenhop': 1, 'Camille Delamarre': 1, 'Steve Boyum': 1, 'John D
ahl': 1, 'Wayne Kramer': 1, 'Brian Robbins': 1, 'Robert Iscove': 1, 'John Herzfel
d': 1}}, 'Brad Pitt': defaultdict(<class 'int'>, {'David Fincher': 2, 'Wolfgang P
etersen': 1, 'Steven Soderbergh': 2, 'Doug Liman': 1, 'Tony Scott': 2, 'David Aye
r': 1, 'Jean-Jacques Annaud': 1, 'Patrick Gilmore': 1, 'Neil Jordan': 1, 'Terrenc
e Malick': 1, 'Andrew Dominik': 2, 'Alejandro G. IÃ±Ã©rritu': 1, 'Angelina Jolie
Pitt': 1, 'Tom DiCillo': 1}}, 'Nicolas Cage': defaultdict(<class 'int'>, {'Jon Tu
rteltaub': 2, 'Mark Steven Johnson': 1, 'John Woo': 2, 'Dominic Sena': 1, 'Michae
l Bay': 1, 'Brian De Palma': 1, 'Oliver Stone': 1, 'David Bowers': 1, 'Brett Ratn

```

```

er': 1, 'Mark Neveldine': 1, 'Brad Silberling': 1, 'Alex Proyas': 1, 'Patrick Lus
sier': 1, 'Francis Ford Coppola': 2, 'John A. Davis': 1, 'Danny Pang': 1, 'Andrew
Niccol': 1, 'Joel Schumacher': 1, 'Barbet Schroeder': 1, 'Simon West': 1, 'Martin
Scorsese': 1, 'Werner Herzog': 1, 'Gore Verbinski': 1, 'Scott Walker': 1, 'Spike
Jonze': 1, 'Vic Armstrong': 1, 'Paul Schrader': 1, 'Mike Figgis': 1, 'David Gordo
n Green': 1}}, 'Justin Timberlake': defaultdict(<class 'int'>, {'Chris Miller':
1, 'Eric Brevig': 1, 'Marco Schnabel': 1, 'Brad Furman': 1, 'Jake Kasdan': 1, 'Cr
aig Brewer': 1, 'Ethan Coen': 1}}, 'Dominic Cooper': defaultdict(<class 'int'>,
{'Duncan Jones': 1, 'Gary Shore': 1, 'Niels Arden Oplev': 1, 'Saul Dibb': 1, '':
1, 'Lone Scherfig': 1, 'Nicholas Hytner': 1}}, 'Bruce Spence': defaultdict(<class
'int'>, {'Michael Apted': 1, 'Steve Oedekerk': 1}}, 'Jennifer Garner': defaultdic
t(<class 'int'>, {'Michael Bay': 1, 'Peter Berg': 1, 'Rob Bowman': 1, 'Peter Hedg
es': 1, 'Gary Winick': 1, 'Ivan Reitman': 1, 'Ricky Gervais': 1, 'Patricia Rigge
n': 1, 'Danny Leiner': 1}}, 'Zack Ward': defaultdict(<class 'int'>, {'Michael Ba
y': 1, 'Zack Ward': 1, 'Marco Kreuzpaintner': 1, '': 1, 'Bob Clark': 1}}, 'Anthon
y Hopkins': defaultdict(<class 'int'>, {'Oliver Stone': 2, 'Joe Johnston': 1, 'Da
rren Aronofsky': 1, 'Martin Brest': 1, 'Lee Tamahori': 1, 'Joel Schumacher': 1,
'Martin Campbell': 1, 'Jon Turteltaub': 1, 'Steven Spielberg': 1, 'Mikael H  fstr
  m': 1, 'Scott Hicks': 1, 'Edward Zwick': 1, 'Roger Donaldson': 1, 'Richard Atte
nborough': 1, 'Jonathan Demme': 1, 'James Ivory': 2, 'Woody Allen': 1, 'Emilio Es
tevez': 1, 'David Lynch': 1}}, 'Robert Pattinson': defaultdict(<class 'int'>, {'D
avid Yates': 1, 'Mike Newell': 1, 'Bill Condon': 1, 'David Slade': 1, 'Chris Weit
z': 1, 'Francis Lawrence': 1, 'Allen Coulter': 1}}, 'Janeane Garofalo': defaulttdi
ct(<class 'int'>, {'Brad Bird': 1, 'Kinka Usher': 1, 'Richard Kelly': 1, 'David D
obkin': 1, 'Risa Bramon Garcia': 1}}, 'Bernie Mac': defaultdict(<class 'int'>,
{'Eric Darnell': 1, 'Malcolm D. Lee': 1, 'Chris Rock': 1, 'Kevin Rodney Sulliva
n': 1, 'Sam Weisman': 1, 'Charles Stone III': 1, 'Terry Zwigoff': 1, 'Lionel C. M
artin': 1, 'Ice Cube': 1, 'Spike Lee': 2}}, 'Robin Williams': defaultdict(<class
'int'>, {'Shawn Levy': 3, 'George Miller': 2, 'Chris Columbus': 2, 'Les Mayfiel
d': 1, 'Vincent Ward': 1, 'Steven Spielberg': 1, 'Tom Shadyac': 1, 'Danny DeVit
o': 1, 'Walt Becker': 1, 'Ken Kwapis': 1, 'Justin Zackham': 1, 'Ron Clements': 1,
'Kirsten Sheridan': 1, 'Barry Levinson': 2, 'Peter Weir': 1, 'Peter Kassovitz':
1, 'Mark Romanek': 1, 'Gus Van Sant': 1, 'Bobcat Goldthwait': 1, 'Patrick Stettne
r': 1, 'David Duchovny': 1, 'James David Pasternak': 1}}, 'Essie Davis': defaulttd
ict(<class 'int'>, {'Lana Wachowski': 1, 'Baz Luhrmann': 1}}, 'Josh Gad': default
dict(<class 'int'>, {'Chris Buck': 1, 'Shawn Levy': 1}}, 'Steve Bastoni': default
dict(<class 'int'>, {'Lana Wachowski': 1}}, 'Kelli Garner': defaultdict(<class 'i
nt'>, {'Hoyt Yeatman': 1, 'Mike Mills': 1, 'Larry Clark': 1}}, 'Matthew Broderic
k': defaultdict(<class 'int'>, {'Steve Hickner': 1, 'Brett Ratner': 1, 'Roger All
ers': 1, 'David Kellogg': 1, 'John Whitesell': 1, 'Ben Stiller': 1, 'Jeff Nathans
on': 1, 'Richard Donner': 1, 'John Badham': 1, 'Alexander Payne': 1, 'Kenneth Lon
ergan': 1}}, 'Seychelle Gabriel': defaultdict(<class 'int'>, {'M. Night Shyamala
n': 1}}, 'Philip Seymour Hoffman': defaultdict(<class 'int'>, {'J.J. Abrams': 1,
'Jan de Bont': 1, 'Anthony Minghella': 2, 'Brett Ratner': 1, 'Mike Nichols': 1,
'Cameron Crowe': 1, 'Bennett Miller': 2, 'Richard Curtis': 1, 'John Hamburg': 1,
'John Patrick Shanley': 1, 'Charlie Kaufman': 1, 'Anton Corbijn': 1, 'Joel Coen':
1, 'Richard Kwietniowski': 1, 'Tamara Jenkins': 1, 'Spike Lee': 1, 'Todd Solond
z': 1, 'Brad Anderson': 1}}, 'Elisabeth Harnois': defaultdict(<class 'int'>, {'Si
mon Wells': 1}}, 'Ty Burrell': defaultdict(<class 'int'>, {'Rob Minkoff': 1, 'Lou
is Leterrier': 1, 'James Bobin': 1, 'Zack Snyder': 1, 'Craig Johnson': 1}}, 'Jada
Pinkett Smith': defaultdict(<class 'int'>, {'Eric Darnell': 2, 'Tom Shadyac': 1,
'Diane English': 1, 'Keenen Ivory Wayans': 1, 'Matty Rich': 1, 'Albert Hughes':
1}}, 'Toby Stephens': defaultdict(<class 'int'>, {'Lee Tamahori': 1, 'Michael Ba
y': 1}}, 'Ed Begley Jr.': defaultdict(<class 'int'>, {'Paul Feig': 1, 'Penny Mars
hall': 1, 'Ben Stassen': 1, 'Paul Schrader': 2, 'Chris Paine': 1}}, 'Bruce Willi
s': defaultdict(<class 'int'>, {'Michael Bay': 1, 'Len Wiseman': 1, 'John Moore':
1, 'John McTiernan': 1, 'Tim Johnson': 1, 'Dean Parisot': 1, 'Jonathan Mostow':
1, 'Antoine Fuqua': 1, 'Barry Levinson': 1, 'Renny Harlin': 1, 'Gregory Hoblit':
1, 'Jon Turteltaub': 1, 'Michael Lehmann': 1, 'Robert Schwentke': 1, 'Harold Beck
er': 1, 'Robert Zemeckis': 1, 'Rob Reiner': 1, 'Richard Donner': 1, 'M. Night Shy

```

```

amalan': 1, 'Kevin Smith': 1, 'Howard Deutch': 1, 'Paul McGuigan': 1, 'Jonathan L
ynn': 1, 'Brian A Miller': 1, 'Wes Anderson': 1, 'Mike Judge': 1, 'Alan Rudolph':
1, 'Quentin Tarantino': 1}}, 'John Michael Higgins': defaultdict(<class 'int'>,
{'Dean Parisot': 1, 'Christopher Guest': 2}), 'Sam Shepard': defaultdict(<class
'int'>, {'Rob Cohen': 1, 'Mateo Gil': 1, 'Terrence Malick': 1}), 'Matt Frewer': d
efaultdict(<class 'int'>, {'Zack Snyder': 1, 'Uwe Boll': 1}), 'Kevin Rankin': def
aultdict(<class 'int'>, {'Ang Lee': 1}), 'Chris Evans': defaultdict(<class 'in
t'>, {'Joe Johnston': 1, 'Tim Story': 1, 'Danny Boyle': 1, 'Joon-ho Bong': 1, 'Ke
vin Munroe': 1, 'Sylvain White': 1, 'Mark Mylod': 1, 'Joel Gallen': 1, 'Ariel Vro
men': 1}), 'Colin Salmon': defaultdict(<class 'int'>, {'Michael Apted': 1, '': 1,
'John Simpson': 1, 'Stuart Hazeldine': 1}), "James D'Arcy": defaultdict(<class 'i
nt'>, {'Peter Weir': 1, 'Renny Harlin': 1}), 'Don Johnson': defaultdict(<class 'i
nt'>, {'': 1, 'Ron Shelton': 1, 'Nick Cassavetes': 1, 'Tom Brady': 1}), 'Mark Ryl
ance': defaultdict(<class 'int'>, {'Steven Spielberg': 1}), 'Matt Damon': default
dict(<class 'int'>, {'Paul Greengrass': 4, 'Neill Blomkamp': 1, 'Ridley Scott':
1, 'Kelly Asbury': 1, 'Terry Gilliam': 1, 'Don Bluth': 1, 'Doug Liman': 1, 'Steve
n Soderbergh': 2, 'Clint Eastwood': 2, 'Robert Redford': 1, 'Bobby Farrelly': 1,
'George Nolfi': 1, 'Stephen Gaghan': 1, 'Billy Bob Thornton': 1, 'Francis Ford Co
ppola': 1, 'Ethan Coen': 1, 'Jeff Schaffer': 1, 'Gus Van Sant': 2, 'Kenneth Loner
gan': 1, 'John Dahl': 1, 'Kevin Smith': 2, 'Charles Ferguson': 1}), 'Jim Parson
s': defaultdict(<class 'int'>, {'Tim Johnson': 1, 'Zach Braff': 1}), 'Salma Haye
k': defaultdict(<class 'int'>, {'Chris Miller': 1, 'Brett Ratner': 1, 'Peter Lor
d': 1, 'Frank Coraci': 1, 'Joachim R  nning': 1, 'Kevin Hooks': 1, 'Mark Christop
her': 1, 'Julie Taymor': 1, 'Allison Anders': 1, 'Dan Ireland': 1}), 'Toby Jone
s': defaultdict(<class 'int'>, {'Steven Spielberg': 1, 'Ron Howard': 1, 'Oliver S
tone': 1, 'John Curran': 1, 'Frank Darabont': 1, 'Douglas McGrath': 1}), 'Daniel
Radcliffe': defaultdict(<class 'int'>, {'Alfonso Cuar  n': 1, 'Chris Columbus':
2, 'Jon M. Chu': 1, 'Paul McGuigan': 1, 'John Boorman': 1, 'James Watkins': 1}),
'Alfre Woodard': defaultdict(<class 'int'>, {'Eric Leighton': 1, 'Michael Tolli
n': 1, 'Liz Friedlander': 1, 'Cathy Malkasian': 1, 'Bille Woodruff': 1, 'Gina Pri
nce-Bythewood': 1, 'Spike Lee': 1, 'John R. Leonetti': 1, 'Bill Benenson': 1}),
'Rupert Grint': defaultdict(<class 'int'>, {'Matt Birch': 2, 'Jonathan Lynn': 1,
'Jeremy Brock': 1}), 'Miguel Ferrer': defaultdict(<class 'int'>, {'Carlos Saldanh
a': 1, 'Les Mayfield': 1, 'John Sayles': 1}), 'Ronny Cox': defaultdict(<class 'in
t'>, {'Paul Verhoeven': 1}), 'Tony Curran': defaultdict(<class 'int'>, {'John McT
iernan': 1, 'Steven Soderbergh': 1, 'Neil Jordan': 1}), 'Jeremy Renner': defaultd
ict(<class 'int'>, {'Tony Gilroy': 1, 'Clark Johnson': 1, 'Tommy Wirkola': 1, 'Be
n Affleck': 1, 'Niki Caro': 1, 'James Gray': 1, 'Juan Carlos Fresnadillo': 1, 'Ka
thryn Bigelow': 1, 'Michael Cuesta': 1}), 'Michael Gough': defaultdict(<class 'in
t'>, {'Joel Schumacher': 2, 'Tim Burton': 2}), 'Clint Howard': defaultdict(<class
'int'>, {'Ron Howard': 1, 'Mike Marvin': 1, 'Jay Russell': 1}), 'Karen Allen': de
faultdict(<class 'int'>, {'Wolfgang Petersen': 1}), 'Suraj Sharma': defaultdict(<
class 'int'>, {'Ang Lee': 1}), 'Demi Moore': defaultdict(<class 'int'>, {'McG':
1, 'Gary Trousdale': 1, 'Barry Levinson': 1, 'Andrew Bergman': 1, 'Jon Cassar':
1, 'Jerry Zucker': 1, 'Derrick Borte': 1}), 'Michael Fassbender': defaultdict(<cl
ass 'int'>, {'Ridley Scott': 2, 'Quentin Tarantino': 1, 'Jimmy Hayward': 1, 'Davi
d Cronenberg': 1, 'Neil Marshall': 1, 'Steve McQueen': 1, 'John Maclean': 1, 'And
rea Arnold': 1, 'James Watkins': 1}), 'Nathan Lane': defaultdict(<class 'int'>,
{'Rob Minkoff': 1, 'Timothy Bj  rklund': 1}), 'Vin Diesel': defaultdict(<class 'i
nt'>, {'David Twohy': 3, 'Breck Eisner': 1, 'Rob Cohen': 1, 'Adam Shankman': 1,
'Brad Bird': 1, 'Mathieu Kassovitz': 1, 'F. Gary Gray': 1, 'Brian Koppelman': 1,
'Ben Younger': 1}), 'Scott Porter': defaultdict(<class 'int'>, {'Lana Wachowski':
1, 'Todd Graff': 1}), 'Shelley Conn': defaultdict(<class 'int'>, {'James L. Brook
s': 1}), 'Morgan Freeman': defaultdict(<class 'int'>, {'Joseph Kosinski': 1, 'Tim
ur Bekmambetov': 1, 'Tom Shadyac': 1, 'Mimi Leder': 1, 'Louis Leterrier': 2, 'Mik
ael Salomon': 1, 'Phil Alden Robinson': 1, 'Lawrence Kasdan': 1, 'Phil Lord': 1,
'George Armitage': 1, 'Rob Reiner': 1, 'Carl Franklin': 1, 'Charles Martin Smit
h': 2, 'David Fincher': 1, 'Lasse Hallstr  m': 1, 'Lee Tamahori': 1, 'Gary Flede
r': 1, 'Frank Darabont': 1, 'Neil LaBute': 1, 'Bruce Beresford': 1, 'Luc Jacque
t': 1}), 'Natalie Portman': defaultdict(<class 'int'>, {'George Lucas': 3, 'Tim B

```



```

urton': 1, 'James McTeigue': 1, 'David Gordon Green': 1, 'Justin Chadwick': 1, 'Mike Nichols': 1, 'Jim Sheridan': 1, 'Ivan Reitman': 1, 'Gavin O'Connor': 1, 'Wayne Wang': 1, 'Matt Williams': 1, 'Darren Aronofsky': 1, 'Kar-Wai Wong': 1}}, 'Alain Delon': defaultdict(<class 'int'>, {'Frédéric Forestier': 1}), 'Noel Fisher': defaultdict(<class 'int'>, {'Jonathan Liebesman': 2, 'Tim Hill': 1}), 'Phaldut Sharma': defaultdict(<class 'int'>, {'Alfonso Cuarón': 1}), 'Jamie Renée Smith': defaultdict(<class 'int'>, {'Roger Donaldson': 1}), 'Stephen Amell': defaultdict(<class 'int'>, {'Dave Green': 1}), 'Tim Blake Nelson': defaultdict(<class 'int'>, {'Josh Trank': 1, 'Andrew Davis': 1, 'Joey Lauren Adams': 1}), 'Dwayne Johnson': defaultdict(<class 'int'>, {'Brad Peyton': 1, 'Adam McKay': 1, 'Brett Ratner': 1, 'Stephen Sommers': 1, 'Peter Berg': 1, 'F. Gary Gray': 1, 'Andrzej Bartkowiak': 1, 'Chuck Russell': 1, 'Jorge Blanco': 1, 'Kevin Bray': 1, 'Rawson Marshall Thurber': 1, 'Andy Fickman': 2, 'Phil Joanou': 1, 'Michael Bay': 1, 'George Tillman Jr.': 1, 'Ric Roman Waugh': 1}), 'Vincent Schiavelli': defaultdict(<class 'int'>, {'Roger Spottiswoode': 1}), 'Heath Ledger': defaultdict(<class 'int'>, {'Roland Emmerich': 1, 'Brian Helgeland': 2, 'Catherine Hardwicke': 1, 'Marc Forster': 1}), 'Kate Winslet': defaultdict(<class 'int'>, {'Robert Schwentke': 1, 'Nancy Meyers': 1, 'Neil Burger': 1, 'Steven Zaillian': 1, 'Stephen Daldry': 1, 'Roman Polanski': 1, 'Michel Gondry': 1, 'John Hillcoat': 1, 'Danny Boyle': 1, 'Todd Field': 1, 'Philip Kaufman': 1, 'John Turturro': 1, 'Richard Eyre': 1, 'Peter Jackson': 1}), 'James Corden': defaultdict(<class 'int'>, {'Rob Letterman': 1, 'Robert B. Weide': 1, 'John Carney': 1}), 'George Peppard': defaultdict(<class 'int'>, {'': 1, 'Jack Smight': 1}), 'Eva Green': defaultdict(<class 'int'>, {'Noam Muro': 1, 'Martin Campbell': 1, 'Benedek Fliegau': 1}), 'Mahadeo Shivraj': defaultdict(<class 'int'>, {'Raja Gosnell': 1}), 'Jacob Tremblay': defaultdict(<class 'int'>, {'Raja Gosnell': 1}), 'Jason Patric': defaultdict(<class 'int'>, {'Jan de Bont': 1, 'Nick Cassavetes': 1, 'Joe Carnahan': 1}), 'Sean Hayes': defaultdict(<class 'int'>, {'Bo Welch': 1}), 'Chazz Palminteri': defaultdict(<class 'int'>, {'Rob Minkoff': 1, 'Keenen Ivory Wayans': 1, 'Chris Weitz': 1, 'Matthew Diamond': 1, 'Paul Mazursky': 1, 'Barrett Esposito': 1, 'Mars Callahan': 1, 'Dave Rodriguez': 1}), 'Oprah Winfrey': defaultdict(<class 'int'>, {'Ron Clements': 1, 'Jonathan Demme': 1, 'Steven Spielberg': 1}), 'Mathew Buck': defaultdict(<class 'int'>, {'Christopher Barnard': 1}), 'Del Zamora': defaultdict(<class 'int'>, {'Peter Chelsom': 1, 'Alex Cox': 1}), 'Djimon Hounsou': defaultdict(<class 'int'>, {'Ridley Scott': 1, 'Stefen Fangmeier': 1, 'Stephen Sommers': 1, 'Shekhar Kapur': 1, 'Jeff Wadlow': 1, 'Julie Taymor': 1, 'Mark Neveldine': 1, 'David E. Talbert': 1}), 'Cary-Hiroyuki Tagawa': defaultdict(<class 'int'>, {'Tim Burton': 1, 'Simon Wincer': 1, 'Bernardo Bertolucci': 1, 'Lasse Hallström': 1, 'Mark L. Lester': 1}), 'Joe Morton': defaultdict(<class 'int'>, {'James Cameron': 1, 'Tom Shadyac': 1, '': 1}), 'Denzel Washington': defaultdict(<class 'int'>, {'Ridley Scott': 1, 'Tony Scott': 4, 'Daniel Espinosa': 1, 'Albert Hughes': 1, 'Jonathan Demme': 2, 'Edward Zwick': 3, 'Baltasar Kormákur': 1, 'Antoine Fuqua': 2, 'Spike Lee': 4, 'Carl Franklin': 1, 'Phillip Noyce': 1, 'Alan J. Pakula': 1, 'Norman Jewison': 1, 'Nick Cassavetes': 1, 'Robert Zemeckis': 1, 'Brett Leonard': 1, 'Richard Attenborough': 1, 'Denzel Washington': 2}), 'Jamie Lee Curtis': defaultdict(<class 'int'>, {'James Cameron': 1, 'Joe Roth': 1, 'Mark Waters': 1, 'Raja Gosnell': 1, 'Andy Fickman': 1, 'John Carpenter': 2, 'Rick Rosenthal': 1, 'Amy Holden Jones': 1}), 'Robert De Niro': defaultdict(<class 'int'>, {'Paul Weitz': 1, 'Des McAnuff': 1, 'Bibo Bergeron': 1, 'Frank Oz': 1, 'Jay Roach': 2, 'Harold Ramis': 2, 'Jon Avnet': 1, 'John Frankenheimer': 1, 'Tony Scott': 1, 'Martin Scorsese': 7, 'Barry Levinson': 3, 'John Herzfeld': 1, 'Nancy Meyers': 1, 'Michael Caton-Jones': 1, 'Peter Segal': 1, 'Scott Mann': 1, 'George Tillman Jr.': 1, 'Martin Brest': 1, 'Nick Hamm': 1, 'Sergio Leone': 1, 'Jon Turteltaub': 1, 'Joel Schumacher': 1, 'Garry Marshall': 1, 'Brian De Palma': 1, 'John Polson': 1, 'Mary McGuckian': 1, 'John Curran': 1, 'Neil Jordan': 1, 'Kirk Jones': 1, 'Jonathan Jakubowicz': 1, 'Rodrigo Cortés': 1, 'Michael Cimino': 1, 'James Mangold': 1, 'Terry Gilliam': 1, 'Francis Ford Coppola': 1, 'Ethan Maniquis': 1, 'Quentin Tarantino': 1, 'Dan Mazer': 1}), 'Vanessa Williams': defaultdict(<class 'int'>, {'Chuck Russell': 1, 'Simon Wincer': 1, 'Gary Halvorson': 1, 'Christopher Erskin': 1, 'George Tillman Jr.': 1, 'Richard Schenkman': 1}), 'Eartha Kitt': defaultdict(<class 'int'>, {'Mark Dindal': 1}), 'Catherin

```

```

e O'Hara": defaultdict(<class 'int'>, {'Spike Jonze': 1, 'Mark Dindal': 1, 'Sam Mendes': 1, 'Tim Burton': 1, 'Dean Parisot': 1, 'Christopher Guest': 1}), 'Josh Hutcherson': defaultdict(<class 'int'>, {'Chris Wedge': 1, 'Brad Peyton': 1, 'Eric Brevig': 1, 'Jesse Dylan': 1, 'Paul Weitz': 1, 'Gabor Csupo': 1, 'Andrea Di Stefano': 1, 'Joseph Kahn': 1, 'Lisa Cholodenko': 1, 'Shari Springer Berman': 1}), 'Jake Busey': defaultdict(<class 'int'>, {'Paul Verhoeven': 1}), 'Abbie Cornish': defaultdict(<class 'int'>, {'Zack Snyder': 1, 'Martin McDonagh': 1, 'Jane Campion': 1}), 'Frances Conroy': defaultdict(<class 'int'>, {'Pitof': 1, 'Anand Tucker': 1, 'Robert Cary': 1}), 'Will Ferrell': defaultdict(<class 'int'>, {'Brad Silberling': 1, 'Jay Roach': 2, 'Adam McKay': 2, 'Josh Gordon': 1, 'Kent Alterman': 1, 'Sean Anders': 1, 'Matthew O'Callaghan': 1, 'Ethan Cohen': 1, 'Marc Forster': 1, 'Todd Phillips': 1, 'Kevin Smith': 1, 'John Fortenberry': 1, 'Nick Gomez': 1, 'Bruce McCulloch': 1, 'Reginald Hudlin': 1, 'Andrew Wilson': 1, 'Matt Piedmont': 1, 'Dan Rush': 1}), 'Ray Winstone': defaultdict(<class 'int'>, {'Ericson Core': 1, 'Pierre Morel': 1, 'Liliana Cavani': 1, 'John Hillcoat': 1, 'Jason Connery': 1, 'Ian Sharp': 1, 'Jonathan Glazer': 1, 'Mora Stephens': 1, 'Antonio Simoncini': 1}), 'Jamie Kennedy': defaultdict(<class 'int'>, {'Lawrence Guterman': 1}), 'Rosario Dawson': defaultdict(<class 'int'>, {'Ron Underwood': 1, 'D.J. Caruso': 1, 'Frank Coraci': 1, 'Frank Miller': 1, 'Chris Columbus': 1, 'Harry Elfont': 1, 'Danny Boyle': 1, 'Craig Bolotin': 1, 'Chris Rock': 1, 'Kris Isacsson': 1, 'Larry Clark': 1}), 'Francesca Capaldi': defaultdict(<class 'int'>, {'Steve Martino': 1}), 'Ben Gazzara': defaultdict(<class 'int'>, {'David Mamet': 1, 'Rowdy Herrington': 1}), 'Jon Lovitz': defaultdict(<class 'int'>, {'Frank Oz': 1, 'Demian Lichtenstein': 1, 'Susan Stroman': 1, 'Dennis Dugan': 1, 'Richard Benjamin': 1, 'Gene Quintano': 1, 'Jerry Rees': 1}), 'Ioan Gruffudd': defaultdict(<class 'int'>, {'Ridley Scott': 1, 'Kevin Lima': 1, 'Alister Grierson': 1, 'Jonathan Newman': 1}), 'Milla Jovovich': defaultdict(<class 'int'>, {'Luc Besson': 1, 'Paul W.S. Anderson': 4, 'Alexander Witt': 1, 'Ben Stiller': 2, 'Russell Mulcahy': 1, 'Kurt Wimmer': 1, 'James McTeigue': 1, 'Michael Winterbottom': 1, 'William A. Graham': 1, 'Olatunde Osunsanmi': 1, 'Richard Linklater': 1}), 'Chris Noth': defaultdict(<class 'int'>, {'Michael Patrick King': 1, '': 1, 'Daniel Sackheim': 1, 'Michael Radford': 1}), 'Frank Welker': defaultdict(<class 'int'>, {'Bibo Bergeron': 1, 'Luis Llosa': 1, 'Maurice Joyce': 1}), 'Hayley Atwell': defaultdict(<class 'int'>, {'Kenneth Branagh': 1, 'Nick Love': 1}), 'Michael Imperioli': defaultdict(<class 'int'>, {'Peter Jackson': 1, 'Brad Anderson': 1, 'Cheryl Dunye': 1, 'Spike Lee': 1}), 'Alexander Gould': defaultdict(<class 'int'>, {'Andrew Stanton': 1, 'Robert Harmon': 1}), 'Orlando Bloom': defaultdict(<class 'int'>, {'Peter Jackson': 1, 'Cameron Crowe': 1, 'JÃ©rÃ©mie Salle': 1}), 'Michael Jeter': defaultdict(<class 'int'>, {'Joe Johnston': 1, 'Anthony Russo': 1}), 'James Franco': defaultdict(<class 'int'>, {'Rupert Wyatt': 1, 'Ryan Murphy': 1, 'Tony Bill': 1, 'John Dahl': 1, 'Evan Goldberg': 1, 'Greg Terneran': 1, 'David Gordon Green': 1, 'Paul Haggis': 1, 'Gus Van Sant': 1, 'Danny Boyle': 1, 'David Raynr': 1, 'Harmony Korine': 1, 'Karen Moncrieff': 1, 'Gia Coppola': 1}), 'Martin Short': defaultdict(<class 'int'>, {'Mark Waters': 1, 'Brenda Chapman': 1, 'John A. Davis': 1, 'John Blanchard': 1}), 'Dennis Quaid': defaultdict(<class 'int'>, {'John Lee Hancock': 2, 'John Moore': 1, 'Lawrence Kasdan': 1, 'Rob Cohen': 1, 'Pete Travis': 1, 'Gary Fleder': 1, 'Gregory Hoblit': 1, 'Philip Kaufman': 1, 'Scott Stewart': 1, 'Sean McNamara': 1, 'Todd Haynes': 1, 'Walter Hill': 1}), 'Holly Hunter': defaultdict(<class 'int'>, {'Brad Bird': 1, 'Nick Hurran': 1, 'Joel Coen': 1, 'Daniel Barnz': 1, 'Richard LaGravenese': 1, 'Jane Campion': 1, '': 1, 'Catherine Hardwicke': 1}), 'Christopher Masterson': defaultdict(<class 'int'>, {'Renny Harlin': 1}), 'Logan Lerman': defaultdict(<class 'int'>, {'Chris Columbus': 1, 'Thor Freudenthal': 1, 'Joel Schumacher': 1, 'James Schamus': 1, 'Wil Shriner': 1, 'Eric Bress': 1, 'Stephen Chbosky': 1}), 'Mei MelanÃ§on': defaultdict(<class 'int'>, {'Brett Ratner': 1}), 'Will Forte': defaultdict(<class 'int'>, {'Phil Lord': 1, 'Cody Cameron': 1, 'Akiva Schaffer': 1}), 'Denis Leary': defaultdict(<class 'int'>, {'Carlos Saldanha': 2}), 'Adam Scott': defaultdict(<class 'int'>, {'Ben Stiller': 1, 'Robert Luketic': 1, 'Alexandre Aja': 1, 'Anand Tucker': 1, 'Michael Dougherty': 1, 'Steve Pink': 1, 'Richard Shepard': 1, 'Nicholas Fackler': 1}), 'Bill Murray': defaultdict(<class 'int'>, {'McG': 1, 'Peter Segal': 1, 'Joe Pytko': 1, 'George Clooney': 1, 'Tim Hill': 1, 'Peter Hewitt': 1, 'We

```

```

s Anderson': 5, 'Bobby Farrelly': 2, 'Gil Kenan': 1, 'Richard Donner': 1, 'Tim Robbins': 1, 'Frank Oz': 1, 'John McNaughton': 1, 'Jon Amiel': 1, 'Sydney Pollack': 1, 'Harold Ramis': 2, 'Theodore Melfi': 1, 'Ivan Reitman': 1, 'Don Kempf': 1, 'Aaron Schneider': 1}), 'Ming-Na Wen': defaultdict(<class 'int'>, {'Tony Bancroft': 1, 'Steven E. de Souza': 1, 'Nelson McCormick': 1}), 'Leonard Nimoy': defaultdict(<class 'int'>, {'Gary Trousdale': 1, 'Robert Wise': 1, 'William Shatner': 1, 'Nicholas Meyer': 2, 'Leonard Nimoy': 2, 'Wayne Thornley': 1}), 'Bella Thorne': defaultdict(<class 'int'>, {'Walt Becker': 1, 'Frank Coraci': 1, 'Miguel Arteta': 1, 'Ari Sandel': 1, 'Tyler Oliver': 1}), 'Adam Sandler': defaultdict(<class 'int'>, {'Dennis Dugan': 3, 'Frank Coraci': 2, 'Peter Segal': 2, 'Adam Shankman': 1, 'Steven Brill': 1, 'James L. Brooks': 1, 'Judd Apatow': 1, 'Sean Anders': 1, 'Paul Thomas Anderson': 1, 'Mike Binder': 1}), 'Haley Joel Osment': defaultdict(<class 'int'>, {'Steven Spielberg': 1, 'Norman Jewison': 1, 'Tim McCanlies': 1, 'Steve Trenbirth': 1, 'Peter Hastings': 1}), 'Marsha Thomason': defaultdict(<class 'int'>, {'Rob Minkoff': 1}), 'Greg Grunberg': defaultdict(<class 'int'>, {'Paul Verhoeven': 1, 'John Whitesell': 1}), 'Curtiss Cook': defaultdict(<class 'int'>, {'Sydney Pollack': 1}), 'Alun Armstrong': defaultdict(<class 'int'>, {'Phillip Noyce': 1}), 'Don Cheadle': defaultdict(<class 'int'>, {'Brian De Palma': 1, 'Mick Jackson': 1, 'Thor Freudenthal': 1, 'Steven Soderbergh': 1, 'Terry George': 1, 'Antoine Fuqua': 1, 'Paul Thomas Anderson': 1, 'Paul Haggis': 1, 'John Michael McDonagh': 1}), 'Anne Hathaway': defaultdict(<class 'int'>, {'Carlos Saldanha': 1, 'Garry Marshall': 2, 'Tommy O'Haver': 1, 'Gary Winick': 1, 'Cory Edwards': 1, 'Julian Jarrold': 1, 'Lone Scherfig': 1, 'Jonathan Demme': 1, 'Mitch Davis': 1, 'Kate Barker-Froyland': 1, '': 1}), 'William Smith': defaultdict(<class 'int'>, {'John Milius': 1, 'Scott Ziehl': 1}), 'Paddy Considine': defaultdict(<class 'int'>, {'Ron Howard': 1, 'Jamie Thraves': 1, 'Pawel Pawlikowski': 1, 'Shane Meadows': 1}), 'Shirley Henderson': defaultdict(<class 'int'>, {'Andrey Konchalovskiy': 1, 'Sally Potter': 1, '': 1}), 'Jeff Bennett': defaultdict(<class 'int'>, {'Kevin Lima': 1}), 'Caroline Dhavernas': defaultdict(<class 'int'>, {'': 1, 'Josh Gordon': 1}), 'Meryl Streep': defaultdict(<class 'int'>, {'Nancy Meyers': 1, 'Curtis Hanson': 1, 'Nora Ephron': 1, 'David Frankel': 2, 'Robert Redford': 1, 'Sydney Pollack': 1, 'Carl Franklin': 1, 'Stephen Frears': 1, 'Stephen Daldry': 1, 'Phyllida Lloyd': 1, 'Robert Altman': 1, 'Fred Zinnemann': 1}), 'Al Pacino': defaultdict(<class 'int'>, {'Steven Soderbergh': 1, 'Dennis Dugan': 1, 'Michael Mann': 1, 'Oliver Stone': 1, 'Francis Ford Coppola': 2, 'Christopher Nolan': 1, 'Harold Becker': 2, 'Jon Avnet': 1, 'Michael Radford': 1, 'Brian De Palma': 1, 'Daniel Algrant': 1, 'Shintaro Shimomura': 1, 'Dan Fogelman': 1}), 'Jon Favreau': defaultdict(<class 'int'>, {'Roger Allers': 1, 'Seth Gordon': 2, 'Peter Billingsley': 1, 'Peyton Reed': 1, 'Peter Berg': 1, 'Hart Bochner': 1, 'Jon Favreau': 1, 'Doug Liman': 1}), 'Bob Hoskins': defaultdict(<class 'int'>, {'Jean-Jacques Annaud': 1, 'Wayne Wang': 1, 'Annabel Jankel': 1, 'Stephen Frears': 1, 'Atom Egoyan': 1, 'Fred Schepisi': 1, 'Nigel Cole': 1, 'Sacha Bennett': 1, 'Shane Meadows': 2}), 'F. Murray Abraham': defaultdict(<class 'int'>, {'John McTiernan': 1, 'Gus Van Sant': 1}), 'Li Gong': defaultdict(<class 'int'>, {'Rob Marshall': 1, 'Pou-Soi Cheang': 1, 'Peter Webber': 1, 'Yimou Zhang': 2, 'Kar-Wai Wong': 1}), 'Amber Stevens West': defaultdict(<class 'int'>, {'Justin Lin': 1}), 'Raymond Cruz': defaultdict(<class 'int'>, {'Andrew Davis': 1}), 'Roy Scheider': defaultdict(<class 'int'>, {'Bob Fosse': 1, 'William Friedkin': 2, 'Jeannot Szwarc': 1, 'Steven Spielberg': 1}), 'Julia Roberts': defaultdict(<class 'int'>, {'Tarsem Singh': 1, 'Richard Donner': 1, 'Garry Marshall': 2, 'Mike Newell': 1, 'Tony Gilroy': 1, 'Steven Soderbergh': 2, 'Chris Columbus': 1, 'Stephen Frears': 1, 'P.J. Hogan': 1, 'Joe Roth': 1, 'Roger Michell': 1, 'Jodie Foster': 1, 'Joel Schumacher': 1, 'Billy Ray': 1, 'Donald Petrie': 1}), 'Anna Kendrick': defaultdict(<class 'int'>, {'Edgar Wright': 1, 'Chris Butler': 1, 'Kirk Jones': 1, 'Elizabeth Banks': 1, 'Jason Moore': 1, 'Richard LaGravenese': 1, 'Joe Swanberg': 1}), 'Larry Miller': defaultdict(<class 'int'>, {'Peter Segal': 1, 'James Dodson': 1}), 'Sarah Michelle Gellar': defaultdict(<class 'int'>, {'Raja Gosnell': 2, 'Paul Bolger': 1, 'Takashi Shimizu': 2, 'Roger Kumble': 1, 'Emily Young': 1, '': 1, 'Mark Tarlov': 1, 'James Toback': 1}), 'Wood Harris': defaultdict(<class 'int'>, {'Pete Travis': 1}), 'Ted Danson': defaultdict(<class 'int'>, {'George A. Romero': 1, 'Ken Kwapis': 1}), 'Kristen Stewart': defaultdict(<

```

```

lass 'int'>, {'Doug Liman': 1, 'Jon Favreau': 1, 'David Fincher': 1, 'Catherine H
ardwicke': 1, 'Walter Salles': 1, 'Sean Penn': 1, 'Bart Freundlich': 1, 'Udayan P
rasad': 1, 'Jon Kasdan': 1, 'Jake Scott': 1, 'Greg Mottola': 1, 'Floria Sigismond
i': 1, 'Richard Glatzer': 1}), 'Seth MacFarlane': defaultdict(<class 'int'>, {'Gu
illermo del Toro': 1}), 'Robert Duvall': defaultdict(<class 'int'>, {'Roger Spott
iswoode': 2, 'Steven Zaillian': 1, 'Jon Turteltaub': 1, 'James Gray': 1, 'Kevin C
ostner': 1, 'Billy Bob Thornton': 1, 'Sidney Lumet': 1, 'Robert Mulligan': 1}),
'Tia Carrere': defaultdict(<class 'int'>, {'Dean DeBlois': 1, 'Penelope Spheeri
s': 1, 'Myles Berkowitz': 1}), 'Elden Henson': defaultdict(<class 'int'>, {'': 1,
'Troy Miller': 1}), 'Roger Rees': defaultdict(<class 'int'>, {'Shawn Levy': 1, 'O
liver Hirschbiegel': 1, 'Robin Budd': 1}), 'Vanessa Redgrave': defaultdict(<class
'int'>, {'Henry Jaglom': 1, 'Gary Winick': 1, 'Roland Emmerich': 1, 'James Ivor
y': 1}), 'Elizabeth Montgomery': defaultdict(<class 'int'>, {'': 1}), 'Quincy Jon
es': defaultdict(<class 'int'>, {'James Algar': 1}), 'Charlize Theron': defaulttdi
ct(<class 'int'>, {'Ron Underwood': 1, 'Karyn Kusama': 1, 'John Frankenheimer':
1, 'James Gray': 1, 'Jason Reitman': 1, 'Patty Jenkins': 1}), 'Michael Emerson':
defaultdict(<class 'int'>, {'Martin Campbell': 1, 'Jay Oliva': 1, 'James Wan':
1}), 'Nissim Renard': defaultdict(<class 'int'>, {'Lucile Hadzihalilovic': 1}),
'Bruce Greenwood': defaultdict(<class 'int'>, {'Roger Donaldson': 1, 'Bruce Beres
ford': 2, 'Frederik Du Chau': 1, 'David Nutter': 1, 'Mark Piznarski': 1, 'Guy Rit
chie': 1, 'Atom Egoyan': 2, 'Kelly Reichardt': 1, 'Andrew Niccol': 1}), 'Sylveste
r Stallone': defaultdict(<class 'int'>, {'Rob Cohen': 1, 'Renny Harlin': 2, 'Mika
el HÄfstrÄm': 1, 'Eric Darnell': 1, 'Peter MacDonald': 1, 'Andrey Konchalovski
y': 1, 'Jim Gillespie': 1, 'Walter Hill': 1, 'Ted Kotcheff': 1, 'Luis Llosa': 1,
'George P. Cosmatos': 1, 'Robert Rodriguez': 1, 'Stephen Kay': 1, 'Ryan Coogler':
1, 'Sylvester Stallone': 1, 'Norman Jewison': 1, 'Bruce Malmuth': 1, 'John G. Avi
ldsen': 1, 'Paul Bartel': 1}), 'Charlie Rowe': defaultdict(<class 'int'>, {'Barry
Cook': 1}), 'Richard Tyson': defaultdict(<class 'int'>, {'Roger Christian': 1, 'I
van Reitman': 1, 'Terron R. Parsons': 1}), 'Brendan Fraser': defaultdict(<class
'int'>, {'Joe Dante': 1, 'Henry Selick': 1, 'Iain Softley': 1, 'Sam Weisman': 1,
'Harold Ramis': 1, 'Cal Brunker': 1, 'Hugh Wilson': 2, 'Roger Kumble': 1, 'Richar
d Benjamin': 1, 'Phillip Noyce': 1, 'Bill Condon': 1}), 'Fergie': defaultdict(<cl
ass 'int'>, {'Rob Marshall': 1}), 'Olivia Williams': defaultdict(<class 'int'>,
{'Kevin Costner': 1, "Thaddeus O'Sullivan": 1}), 'Adam Goldberg': defaultdict(<cl
ass 'int'>, {'George Miller': 1, 'Ron Howard': 1, 'Bibo Bergeron': 1, 'William Br
ent Bell': 1, 'D.J. Caruso': 1}), 'Bob Neill': defaultdict(<class 'int'>, {'Anton
y Hoffman': 1}), 'Mia Farrow': defaultdict(<class 'int'>, {'Luc Besson': 1, 'Nadi
a Tass': 1}), 'Pedro ArmendÄriz Jr.': defaultdict(<class 'int'>, {'Jacques Perri
n': 1}), 'David Oyelowo': defaultdict(<class 'int'>, {'Peter Hyams': 1, 'Anthony
Hemingway': 1, 'Ava DuVernay': 1, 'J.C. Chandor': 1, 'Kevin Macdonald': 1, 'Jerry
Jameson': 1}), 'Sasha Roiz': defaultdict(<class 'int'>, {'Paul W.S. Anderson':
1}), 'Sariann Monaco': defaultdict(<class 'int'>, {'AndrÄs Couturier': 1}), 'Ste
ve Coogan': defaultdict(<class 'int'>, {'Yarrow Cheney': 1, 'Frank Oz': 1, 'Steph
en Frears': 1, 'Andrew Fleming': 1, 'Michael Winterbottom': 1}), 'Jason Flemyng':
defaultdict(<class 'int'>, {'Stephen Norrington': 1, '': 1, 'Alexandre Aja': 1,
'Don Mancini': 1, 'Oleg Stepchenko': 1, 'Bob Spiers': 1, 'Jonathan English': 1}),
'Steve Carell': defaultdict(<class 'int'>, {'Pierre Coffin': 2, 'Kyle Balda': 1,
'Jay Roach': 1, 'Judd Apatow': 1, 'Joe Nussbaum': 1, 'Lorene Scafaria': 1, 'Jonat
han Dayton': 1, 'Peter Sollett': 1, 'Nat Faxon': 1}), 'Ariana Richards': defaultd
ict(<class 'int'>, {'Steven Spielberg': 1, 'Patrick Read Johnson': 1}), 'Jerry St
iller': defaultdict(<class 'int'>, {'Adam Shankman': 1, 'Lawrence Kasanoff': 1,
'Eric Bross': 1}), 'Harold Perrineau': defaultdict(<class 'int'>, {'': 1, 'Bob Ra
felson': 1, 'Malcolm D. Lee': 2, 'Fina Torres': 1}), 'Nicholas Lea': defaultdict
(<class 'int'>, {'Martin Campbell': 1}), 'Loretta Devine': defaultdict(<class 'in
t'>, {'Bill Condon': 1, 'John Ottman': 1, 'Michael Schultz': 1}), 'Adam Arkin': d
efaultdict(<class 'int'>, {'': 1, 'Diane Keaton': 1}), 'Ayelet Zurer': defaultdic
t(<class 'int'>, {'Steven Spielberg': 1}), 'Tom Selleck': defaultdict(<class 'in
t'>, {'Robert Luketic': 1, 'Frank Oz': 1, 'Leonard Nimoy': 1}), 'Julia Ormond': d
efaultdict(<class 'int'>, {'Jerry Zucker': 1, 'Patricia Rozema': 1}), 'Bai Ling':
defaultdict(<class 'int'>, {'Andy Tennant': 1}), 'Connie Nielsen': defaultdict(<c

```

```

lass 'int'>, {'Paul W.S. Anderson': 1, 'William Friedkin': 1, 'John McTiernan':
1, 'McG': 1, 'Vic Sarin': 1}), 'Christopher Meloni': defaultdict(<class 'int'>,
{'David Pastor': 1, 'Lana Wachowski': 1, 'David Wain': 1}), 'Peter Coyote': defau
ltdict(<class 'int'>, {'Barry Levinson': 1, 'Brian De Palma': 1, 'Michael Polis
h': 1}), 'Raven-SymonÃ@': defaultdict(<class 'int'>, {'Steve Carr': 1}), 'Mhairi
Calvey': defaultdict(<class 'int'>, {'Mel Gibson': 1}), 'Albert Brooks': defaultd
ict(<class 'int'>, {'David Silverman': 1, 'Joe Dante': 1}), 'Martin Landau': defa
ultdict(<class 'int'>, {'Frank Darabont': 1, 'Rob Bowman': 1, 'Joseph L. Mankiewi
cz': 1, 'Tim Burton': 1, 'Shane Acker': 1, 'Steve Barron': 1, 'George Stevens':
1, 'Ronald Neame': 1, 'Daniel Davila': 1}), 'David Gant': defaultdict(<class 'in
t'>, {'Jean-Jacques Annaud': 1}), 'Rory Culkin': defaultdict(<class 'int'>, {'M.
Night Shyamalan': 1, 'Arie Posin': 1, 'Jacob Aaron Estes': 1}), 'Rupert Everett':
defaultdict(<class 'int'>, {'Andrew Adamson': 1}), 'Tim Conway': defaultdict(<cla
ss 'int'>, {'Paul Tibbitt': 1, '': 1}), 'Lili Taylor': defaultdict(<class 'int'>,
{'Ron Howard': 1}), 'Nicole Beharie': defaultdict(<class 'int'>, {'': 1, 'Stefan
C. Schaefer': 1}), 'Sanaa Lathan': defaultdict(<class 'int'>, {'Paul W.S. Anderso
n': 1, 'Stephen Norrington': 1}), 'LeVar Burton': defaultdict(<class 'int'>, {'Jo
nathan Frakes': 2, 'David Carson': 1, '': 1}), 'David Hyde Pierce': defaultdict(<
class 'int'>, {'Mike Nichols': 1, 'Nick Tomnay': 1}), 'Jeremy Irvine': defaultdic
t(<class 'int'>, {'Steven Spielberg': 1, 'Jonathan Teplitzky': 1, 'Roland Emmeric
h': 1, 'Ol Parker': 1}), 'Michael Biehn': defaultdict(<class 'int'>, {'James Came
ron': 3, 'William Friedkin': 1, 'Christian Duguay': 1, 'Jonathan Frakes': 1, 'Geo
rge P. Cosmatos': 1, 'Brian Trenchard-Smith': 1, 'Xavier Gens': 1}), 'Frank Lange
lla': defaultdict(<class 'int'>, {'Oliver Stone': 1, 'Richard Kelly': 1, 'Frankli
n J. Schaffner': 1, 'Jake Schreier': 1}), 'Gary Cole': defaultdict(<class 'int'>,
{'Betty Thomas': 1, 'Tim Hill': 1, 'Rawson Marshall Thurber': 1, 'Ben Falcone':
1, 'Sam Raimi': 1, 'Mike Judge': 1, 'Mitch Davis': 1}), 'Theo James': defaultdict
(<class 'int'>, {'MÃns MÃrlind': 1}), 'James Martin Kelly': defaultdict(<class
'int'>, {'Adam Shankman': 1}), 'LL Cool J': defaultdict(<class 'int'>, {'John McT
iernan': 1, 'Renny Harlin': 2, 'Wayne Wang': 1, 'Daisy von Scherler Mayer': 1, 'M
ichael Rymer': 1}), 'Talisa Soto': defaultdict(<class 'int'>, {'Wych Kaosayanand
a': 1}), 'Ron Howard': defaultdict(<class 'int'>, {'Bobby Farrelly': 1}), 'Lea Mi
chele': defaultdict(<class 'int'>, {'Will Finn': 1, 'Kevin Tancharoen': 1}), 'Cha
rlotte Rampling': defaultdict(<class 'int'>, {'Michael Caton-Jones': 1, 'Norman J
ewison': 1, 'Sidney Lumet': 1, 'FranÃois Ozon': 1, 'Hans Petter Moland': 1, 'Jea
n-Jacques Mantello': 1}), 'Roxanne McKee': defaultdict(<class 'int'>, {'Renny Har
lin': 1}), 'Vincent Pastore': defaultdict(<class 'int'>, {'Joseph Ruben': 1, 'Rob
Pritts': 1, 'James Melkonian': 1, 'Danny Provenzano': 1}), 'Max Minghella': defau
ltdict(<class 'int'>, {'Alejandro AmenÃ¡bar': 1, 'Chris Gorak': 1}), 'Jenna Fisch
er': defaultdict(<class 'int'>, {'Bobby Farrelly': 1, 'Brian Koppelman': 1, 'Bob
Odenkirk': 1}), 'Michael Raymond-James': defaultdict(<class 'int'>, {'Craig Gille
spie': 1}), 'Rufus Sewell': defaultdict(<class 'int'>, {'Timur Bekmambetov': 1,
'Chuck Russell': 1, 'Christian Duguay': 1, 'Alex Proyas': 1, 'Neil Burger': 1, 'E
dward Hall': 1}), 'Will Patton': defaultdict(<class 'int'>, {'Jon Amiel': 1}), 'H
ector Elizondo': defaultdict(<class 'int'>, {'': 1, 'Robert Butler': 1, 'Garry Ma
rshall': 1, 'Jesse Dylan': 1}), 'Colin Firth': defaultdict(<class 'int'>, {'Doug
Lefler': 1, 'Phyllida Lloyd': 1, 'Beeban Kidron': 1, 'Richard Curtis': 1, 'Anthon
y Minghella': 1, 'Sharon Maguire': 1, 'Kirk Jones': 1, 'Dennie Gordon': 1, 'Atom
Egoyan': 1, 'Rowan Joffe': 1, 'Tom Hooper': 1, 'Oliver Parker': 1, 'Stephan Ellio
tt': 1, 'Anand Tucker': 1, 'Tom Ford': 1}), 'Rami Malek': defaultdict(<class 'in
t'>, {'Scott Waugh': 1, 'Spike Lee': 1}), 'Goran Visnjic': defaultdict(<class 'in
t'>, {'Chris Wedge': 1, 'Griffin Dunne': 1, 'Mike Mills': 1}), 'Judge Reinhold':
defaultdict(<class 'int'>, {'Michael Lembeck': 1, 'John Pasquin': 1, 'Martin Bres
t': 1}), 'Finn Wittrock': defaultdict(<class 'int'>, {'Angelina Jolie Pitt': 1}),
'Clint Eastwood': defaultdict(<class 'int'>, {'Clint Eastwood': 10, 'Robert Loren
z': 1, 'Joshua Logan': 1, 'Don Siegel': 1, 'Ted Post': 1, 'Sergio Leone': 2}), 'D
elroy Lindo': defaultdict(<class 'int'>, {'John Woo': 1, 'Taylor Hackford': 1, 'B
arry Sonnenfeld': 1, 'David Webb Peoples': 1}), 'Sayed Badreya': defaultdict(<cla
ss 'int'>, {'Larry Charles': 1}), 'QuvenzhanÃ© Wallis': defaultdict(<class 'in
t'>, {'Will Gluck': 1, 'Steve McQueen': 1, 'Ben Zeitlin': 1}), 'Lea Thompson': d

```

```

defaultdict(<class 'int'>, {'John Milius': 1, 'Robert Zemeckis': 3, 'Willard Huyck': 1, 'Alex Ranarivelo': 1}), 'Maura Tierney': defaultdict(<class 'int'>, {'Mike Nichols': 1, 'Tom Shadyac': 1}), 'Melina Kanakaredes': defaultdict(<class 'int'>, {'Renny Harlin': 1}), 'Pamela Reed': defaultdict(<class 'int'>, {'Taylor Hackford': 1}), 'Michael Rapaport': defaultdict(<class 'int'>, {'Nora Ephron': 1, 'Paul Feig': 1, 'Marc Bennett': 1, 'Hal Haberman': 1}), 'Caitlin Stasey': defaultdict(<class 'int'>, {'Stuart Beattie': 1}), 'Ian McNeice': defaultdict(<class 'int'>, {'Roman Polanski': 1}), 'Michael Kelly': defaultdict(<class 'int'>, {'Baltasar Kormákur': 1, 'Marc Lawrence': 1, 'Josh Trank': 1, 'Peter Stebbings': 1}), 'Michael Smiley': defaultdict(<class 'int'>, {'Tom Tykwer': 1, 'Edgar Wright': 1, 'Ben Wheatley': 1}), 'Verne Troyer': defaultdict(<class 'int'>, {'Jay Roach': 1}), 'Wayne Knight': defaultdict(<class 'int'>, {'Steven Spielberg': 1}), 'Si Won Choi': defaultdict(<class 'int'>, {'Daniel Lee': 1}), 'Shawna Waldron': defaultdict(<class 'int'>, {'Rob Reiner': 1}), 'Craig Stark': defaultdict(<class 'int'>, {'Quentin Tarantino': 1}), 'Eric Dane': defaultdict(<class 'int'>, {'David Frankel': 1, 'Steve Antin': 1, 'Rich Cowan': 1}), 'Jill Hennessy': defaultdict(<class 'int'>, {'Walt Becker': 1}), 'Patrick Fugit': defaultdict(<class 'int'>, {'David Fincher': 1, 'Peter Kosminsky': 1, 'Jonas Åkerlund': 1}), 'Izabella Scorupco': defaultdict(<class 'int'>, {'Martin Campbell': 1}), 'Daniel von Bargen': defaultdict(<class 'int'>, {'Simon West': 1}), 'Natascha McElhone': defaultdict(<class 'int'>, {'Peter Weir': 1, 'William Malone': 1}), 'Anjelica Huston': defaultdict(<class 'int'>, {'Steve Carr': 1, 'Barry Sonnenfeld': 1, 'Andy Tennant': 1, 'Bob Rafelson': 1, 'Mira Nair': 1, 'Stewart Raffill': 1, 'Clark Gregg': 1, 'Vincent Gallo': 1}), 'Carol Ann Susi': defaultdict(<class 'int'>, {'Lawrence Guterman': 1}), 'Dorian Missick': defaultdict(<class 'int'>, {'Marc Lawrence': 1, '': 1, 'Qasim Basir': 1}), 'Taylor Lautner': defaultdict(<class 'int'>, {'Adam Shankman': 1, 'Frank Coraci': 1, 'Robert Rodriguez': 1}), 'Ki Hong Lee': defaultdict(<class 'int'>, {'Wes Ball': 2}), 'Billy Burke': defaultdict(<class 'int'>, {'Jay Russell': 1, 'Gregory Hoblit': 1, 'Robert Towne': 1, 'Charles Matthau': 1, 'David F. Sandberg': 1, '': 1}), 'Misha Gabriel Hamilton': defaultdict(<class 'int'>, {'Kenny Ortega': 1}), 'Estella Warren': defaultdict(<class 'int'>, {'David McNally': 1, 'Wayne Kramer': 1}), 'Jennifer Saunders': defaultdict(<class 'int'>, {'Henry Selick': 1}), 'Zooey Deschanel': defaultdict(<class 'int'>, {'M. Night Shyamalan': 1, 'Garth Jennings': 1, 'Barry Sonnenfeld': 1, 'Lawrence Kasdan': 1, 'Ed Decter': 1, 'Michael Clancy': 1, 'Jesse Peretz': 1, 'Adam Rapp': 1, 'David Gordon Green': 1}), 'Snoop Dogg': defaultdict(<class 'int'>, {'Todd Phillips': 1, 'Alan Jacobs': 1}), 'Jim Belushi': defaultdict(<class 'int'>, {'Brian Levant': 1, 'Roman Polanski': 1, 'John Pasquin': 1, 'Oliver Stone': 1}), 'James Babson': defaultdict(<class 'int'>, {'Guillermo del Toro': 1}), 'Ivana Milicevic': defaultdict(<class 'int'>, {'John Woo': 1, 'Charles Robert Carner': 1, 'Jeff Franklin': 1}), 'Emma Watson': defaultdict(<class 'int'>, {'Sam Fell': 1, 'Alejandro Amenábar': 1}), 'Bianca Kajlich': defaultdict(<class 'int'>, {'': 1, 'Ruben Fleischer': 1}), 'Romany Malco': defaultdict(<class 'int'>, {'Kevin Donovan': 1, 'Tim Story': 1}), 'Peter Greene': defaultdict(<class 'int'>, {'Geoff Murphy': 1, 'Les Mayfield': 1, 'Chuck Russell': 1, 'Ash Baron-Cohen': 1}), 'Colm Feore': defaultdict(<class 'int'>, {'Kenneth Branagh': 1, 'Rob Marshall': 1, 'Kasi Lemmons': 1, 'Erik Canuel': 1}), 'Gary Dourdan': defaultdict(<class 'int'>, {'Jean-Pierre Jeunet': 1, 'Gary Fleder': 1, 'Salim Akil': 1}), 'Tate Donovan': defaultdict(<class 'int'>, {'Antoine Fuqua': 1}), 'Isaac Hempstead Wright': defaultdict(<class 'int'>, {'Graham Annable': 1, 'Nick Murphy': 1}), 'Eileen Brennan': defaultdict(<class 'int'>, {'John Pasquin': 1, 'Howard Zeff': 1, 'Victor Salva': 1, 'George Roy Hill': 1}), 'Ryan Gosling': defaultdict(<class 'int'>, {'Ruben Fleischer': 1, 'Glenn Ficarra': 1, 'Barbet Schroeder': 1, 'Boaz Yakin': 1, 'Nick Cassavetes': 1, 'Adam McKay': 1, 'George Clooney': 1, 'Andrew Jarecki': 1, 'Nicolas Winding Refn': 2, 'Derek Cianfrance': 2, 'Craig Gillespie': 1, 'Henry Bean': 1, 'Ryan Fleck': 1, 'Alex Smith': 1}), 'Olivia Wilde': defaultdict(<class 'int'>, {'Harold Ramis': 1, 'Paul Haggis': 1, 'Jessie Nelson': 1, 'David Gelb': 1, 'Joe Swanberg': 1, 'Hans Canosa': 1}), 'Steven Anthony Lawrence': defaultdict(<class 'int'>, {'Donald Petrie': 1}), 'Blair Brown': defaultdict(<class 'int'>, {'Clark Johnson': 1}), 'Cedric the Entertainer': defaultdict(<class 'int'>, {'Joel Coen': 1, 'Darnell Martin': 1}), 'Bojana Novakovic': defaultdict(<class 'int'>, {'': 1})

```

```
t(<class 'int'>, {'Martin Campbell': 1, 'Sam Raimi': 1, 'John Erick Dowdle': 1}),
'John Kapelos': defaultdict(<class 'int'>, {'Peter Hyams': 1}), 'Sunny Mabrey': d
efaultdict(<class 'int'>, {'Lee Tamahori': 1}), 'Josh Wingate': defaultdict(<clas
s 'int'>, {'Scott Stewart': 1}), 'Sean Pertwee': defaultdict(<class 'int'>, {'Pau
l W.S. Anderson': 1, 'Danny Cannon': 1, 'Martyn Pick': 1, 'Gary Sinyor': 1}), 'Cl
ifton Collins Jr.': defaultdict(<class 'int'>, {'Rod Lurie': 1, 'William Dear':
1, 'Christine Jeffs': 1, 'Benni Diez': 1, 'Brandon Trost': 1}), 'Robert Forster':
defaultdict(<class 'int'>, {'Walter Hill': 1, 'Bobby Farrelly': 1, 'Hyung-rae Shi
m': 1, 'Gary Nelson': 1}), 'Matt Bomer': defaultdict(<class 'int'>, {'Akiva Golds
man': 1, 'Andrew Niccol': 1, 'Jonathan Liebesman': 1, 'Gregory Jacobs': 1}), 'Mar
c Blucas': defaultdict(<class 'int'>, {'Brian Robbins': 1, 'Michael Tollin': 1,
'Robby Henson': 1}), 'Dougray Scott': defaultdict(<class 'int'>, {'Walter Salle
s': 1}), 'Kelly Preston': defaultdict(<class 'int'>, {'Stephen Herek': 1, 'Steve
Oedekerk': 1, 'Julie Anne Robinson': 1, 'Ivan Reitman': 1}), 'Odeya Rush': defaul
tdict(<class 'int'>, {'Rob Letterman': 1}), 'Nate Richert': defaultdict(<class 'i
nt'>, {'': 1}), 'Jon Heder': defaultdict(<class 'int'>, {'Mark Waters': 1, 'Todd
Phillips': 1, 'Lijun Sun': 1, 'Jared Hess': 1}), 'Taylor Negron': defaultdict(<cl
ass 'int'>, {'Brian Levant': 1, 'Amy Heckerling': 1, 'Alex Zamm': 1, 'Lance Kawa
s': 1}), 'Robert Baker': defaultdict(<class 'int'>, {'George Clooney': 1}), 'Barr
y Corbin': defaultdict(<class 'int'>, {'': 1, 'Tommy Lee Jones': 1, 'Steve Rash':
1, 'Robert M. Young': 1, 'Kirk Loudon': 1}), 'Jesse McCartney': defaultdict(<clas
s 'int'>, {'Tim Hill': 1, 'Bradley Parker': 1}), 'Marlon Brando': defaultdict(<cl
ass 'int'>, {'Richard Donner': 1, 'John Frankenheimer': 1, 'Elia Kazan': 2}), 'Al
yson Hannigan': defaultdict(<class 'int'>, {'Jesse Dylan': 1, 'Jon Hurwitz': 1,
'J.B. Rogers': 1, 'Aaron Seltzer': 1, 'Robert Iscove': 1, 'Alan Cohn': 1, 'Paul W
eitz': 1, 'Bob Saget': 1}), 'Mila Kunis': defaultdict(<class 'int'>, {'Shawn Lev
y': 1, 'Seth MacFarlane': 1, 'Will Gluck': 1, 'John Moore': 1, 'Nicholas Stolle
r': 1, 'Guillaume Canet': 1, 'Jon Lucas': 1, "Tommy O'Haver": 1}), 'Eric Idle': d
efaultdict(<class 'int'>, {'Brad Silberling': 1, 'Marc F. Adler': 1, 'Arthur Hill
er': 1, 'Terry Gilliam': 1}), 'Erika Christensen': defaultdict(<class 'int'>, {'R
obert Schwentke': 1, 'Bob Dolman': 1, 'John Polson': 1}), 'David Carradine': defa
ultdict(<class 'int'>, {'Quentin Tarantino': 1, 'Jason Friedberg': 1, 'Steve Carv
er': 1}), 'Christopher Judge': defaultdict(<class 'int'>, {'': 1, 'David Hewlet
t': 1}), 'Xander Berkeley': defaultdict(<class 'int'>, {'Tom Dey': 1}), 'Madeline
Carroll': defaultdict(<class 'int'>, {'Mark Waters': 1, 'Brian Levant': 1, 'Joshua
Michael Stern': 1, 'Simon West': 1, 'Rob Reiner': 1}), 'Charles S. Dutton': def
aultdict(<class 'int'>, {'David Fincher': 1}), 'Andrea Corr': defaultdict(<class
'int'>, {'Alan Parker': 1}), 'Tom Wilkinson': defaultdict(<class 'int'>, {'Stephe
n Hopkins': 1, 'Gil Junger': 1, 'Ken Scott': 1, 'John Madden': 3, 'Tony Gilroy':
1, 'Alejandro Monteverde': 1, 'Scott Derrickson': 1, 'Gillian Armstrong': 1, 'Pet
er Cattaneo': 1, 'Todd Field': 1}), 'Julie Walters': defaultdict(<class 'int'>,
{'Paul King': 1, 'John Crowley': 1, 'Stephen Daldry': 1}), 'Eddie Redmayne': defa
ultdict(<class 'int'>, {'Shekhar Kapur': 1, 'James Marsh': 1, 'Simon Curtis': 1,
'Tom Kalin': 1}), 'Billy Campbell': defaultdict(<class 'int'>, {'Ron Maxwell':
1}), 'Stephen Root': defaultdict(<class 'int'>, {'Karey Kirkpatrick': 1, 'Kevin S
mith': 1, 'Ridley Scott': 1, 'Callie Khouri': 1, 'Jeff Lowell': 1}), 'Sophia Myle
s': defaultdict(<class 'int'>, {'Jonathan Frakes': 1, 'Len Wiseman': 2}), 'Denis
Lavant': defaultdict(<class 'int'>, {'Jean-Pierre Jeunet': 1, 'Christophe Ali':
1}), 'James Mason': defaultdict(<class 'int'>, {'Stanley Kubrick': 1, 'Anthony Ma
nn': 1, 'Richard Fleischer': 1}), 'Norman Reedus': defaultdict(<class 'int'>, {'G
uillermo del Toro': 1, 'Christian Alvart': 1, 'Davis Guggenheim': 1, 'Scott Kalve
rt': 1, 'Troy Duffy': 2}), 'Todd Stashwick': defaultdict(<class 'int'>, {'Anthony
Russo': 1}), 'Margot Kidder': defaultdict(<class 'int'>, {'Richard Lester': 2}),
'Todd Giebenhain': defaultdict(<class 'int'>, {'Martin Brest': 1}), 'Kirsten Duns
t': defaultdict(<class 'int'>, {'Don Bluth': 1, 'Joe Dante': 1, 'Sofia Coppola':
2, 'Richard Loncraine': 1, 'Jeff Nichols': 1, 'John Stockwell': 1, 'Peyton Reed':
1, 'Michael Patrick Jann': 1, '': 1, 'Leslye Headland': 1}), 'Quentin Tarantino':
defaultdict(<class 'int'>, {'Robert Rodriguez': 3, 'Morgan Spurlock': 1, 'Quentin
Tarantino': 1}), 'Jennifer Ehle': defaultdict(<class 'int'>, {'Kathryn Bigelow':
1, 'Sam Taylor-Johnson': 1, "Gavin O'Connor": 1, 'Mike Binder': 1, 'Shana Feste':
```



```

1)), 'Alice Greczyn': defaultdict(<class 'int'>, {'Jay Chandrasekhar': 1}), 'Matt
Price': defaultdict(<class 'int'>, {'Milos Forman': 1}), 'Kasia Smutniak': defaul
tdict(<class 'int'>, {'Pierre Morel': 1}), 'Jaime King': defaultdict(<class 'in
t'>, {'Paul Hunter': 1, 'Dewey Nicks': 1}), 'Rob Paulsen': defaultdict(<class 'in
t'>, {'Steve Oedekerk': 1, 'Guillaume Ivernel': 1}), 'Kathleen Freeman': defaultd
ict(<class 'int'>, {'Andrew Adamson': 1}), 'Lupe Ontiveros': defaultdict(<class
'int'>, {'James L. Brooks': 1, 'Miguel Arteta': 1}), 'Joel Courtney': defaultdict
(<class 'int'>, {'J.J. Abrams': 1, '': 1}), 'Bill Duke': defaultdict(<class 'in
t'>, {'Brian Helgeland': 1, 'Mark L. Lester': 1, 'Craig R. Baxley': 1}), 'Dylan W
alsh': defaultdict(<class 'int'>, {'Frank Marshall': 1}), 'Fann Wong': defaultdic
t(<class 'int'>, {'David Dobkin': 1}), 'Bernard Hill': defaultdict(<class 'int'>,
{'': 1}), 'Scoot McNairy': defaultdict(<class 'int'>, {'Angela Robinson': 1, 'Gar
eth Edwards': 1}), 'Olivier Martinez': defaultdict(<class 'int'>, {'Adrian Lyne':
1, 'Jean-Paul Rappeneau': 1, 'Katja von Garnier': 1, 'Bigas Luna': 1}), 'Alex Pet
tyfer': defaultdict(<class 'int'>, {'D.J. Caruso': 1, 'Geoffrey Sax': 1, 'Lee Dan
iels': 1, 'Shana Feste': 1, 'Daniel Barnz': 1}), 'Viggo Mortensen': defaultdict(<
class 'int'>, {'Ridley Scott': 1, 'David Cronenberg': 2, 'AgustÃ\xadn DÃ\xadaz Ya
nes': 1, 'John Hillcoat': 1, 'Ed Harris': 1, 'Vicente Amorim': 1, 'Tony Goldwyn':
1, 'David Oelhoffen': 1, 'Kevin Spacey': 1, 'Renny Harlin': 1, 'Gregory Widen':
1}), 'Deborah Kara Unger': defaultdict(<class 'int'>, {'David Fincher': 1, 'Geoff
rey Sax': 1}), 'Radha Mitchell': defaultdict(<class 'int'>, {'Christophe Gans':
1, 'Michael J. Bassett': 1, 'Roger Spottiswoode': 1, 'Breck Eisner': 1, 'Joel Sch
umacher': 1, 'Marc Forster': 1, 'Emma-Kate Croghan': 1}), 'Matt Walsh': defaulttdi
ct(<class 'int'>, {'Steven Quale': 1, 'Jay Duplass': 1}), 'Louis Lombardi': defau
ltdict(<class 'int'>, {'John Landis': 1, 'Luke Greenfield': 1, 'James Foley':
1}), 'Armin Mueller-Stahl': defaultdict(<class 'int'>, {'Mimi Leder': 1}), 'Mike
Starr': defaultdict(<class 'int'>, {'Steven Seagal': 1, 'Woody Allen': 1, 'Michae
l Schultz': 1, 'Brian Caunter': 1}), 'Amber Valletta': defaultdict(<class 'int'>,
{'Garry Marshall': 1}), 'ShÃ\xad Kosugi': defaultdict(<class 'int'>, {'James McTeigu
e': 1}), 'David Jason': defaultdict(<class 'int'>, {'': 1}), 'Muse Watson': defau
ltdict(<class 'int'>, {'Richard Donner': 1, 'Jay Roach': 1, 'Danny Cannon': 1, 'J
im Gillespie': 1, 'Ed Gass-Donnelly': 1}), 'J.D. Evermore': defaultdict(<class 'i
nt'>, {'Andrew Niccol': 1}), 'Grey Damon': defaultdict(<class 'int'>, {'': 1}),
'Christina Milian': defaultdict(<class 'int'>, {'Stephen Herek': 1}), 'Paul Brook
e': defaultdict(<class 'int'>, {'Luc Besson': 1}), 'Mia Kirshner': defaultdict(<c
lass 'int'>, {'Costa-Gavras': 1}), 'Julie Benz': defaultdict(<class 'int'>, {'':
1, 'Lexi Alexander': 1, 'David Hackl': 1, 'Gary Hardwick': 1, 'Darren Stein':
1}), 'Ian Ziering': defaultdict(<class 'int'>, {'Tony Scott': 1, 'Anthony C. Ferr
ante': 1}), 'Alden Ehrenreich': defaultdict(<class 'int'>, {'Richard LaGravenes
e': 1, 'Francis Ford Coppola': 1}), 'Chris Klein': defaultdict(<class 'int'>, {'A
ndrzej Bartkowiak': 1}), 'Tobias Menzies': defaultdict(<class 'int'>, {'': 1}),
'Lauren Holly': defaultdict(<class 'int'>, {'Renny Harlin': 1, 'Peter Farrelly':
1}), 'Michael O'Neill': defaultdict(<class 'int'>, {'Steven Soderbergh': 1}), 'P.
J. Soles': defaultdict(<class 'int'>, {'Chuck Bowman': 1}), 'Corinna Harney': def
aultdict(<class 'int'>, {'Jerry Zucker': 1}), 'Nandita Das': defaultdict(<class
'int'>, {'Deepa Mehta': 1}), 'Charlie Korsmo': defaultdict(<class 'int'>, {'Warre
n Beatty': 1}), 'Mark Margolis': defaultdict(<class 'int'>, {'John McTiernan': 1,
'Matt Reeves': 1, 'Darren Aronofsky': 2}), 'Rosie Perez': defaultdict(<class 'in
t'>, {'Penny Marshall': 1, 'Patricia Cardoso': 1}), 'Donatas Banionis': defaulttdi
ct(<class 'int'>, {'Andrei Tarkovsky': 1}), 'Joseph Gilgun': defaultdict(<class
'int'>, {'Brad Furman': 1, 'James Mather': 1, 'Daniel Barber': 1}), 'Laurence Oli
vier': defaultdict(<class 'int'>, {'Terence Young': 1, 'Franklin J. Schaffner':
1, 'Alfred Hitchcock': 1}), 'Jonathan Winters': defaultdict(<class 'int'>, {'Bria
n Levant': 1, 'Russell Mulcahy': 1, 'Stanley Kramer': 1}), 'Regina Hall': default
dict(<class 'int'>, {'David Zucker': 1}), 'Candice Bergen': defaultdict(<class 'i
nt'>, {'Donald Petrie': 1, 'Richard Attenborough': 1}), 'Bradley Whitford': defau
ltdict(<class 'int'>, {'Joel Schumacher': 1}), 'Natasha Lyonne': defaultdict(<cla
ss 'int'>, {'Keenen Ivory Wayans': 1, 'Woody Allen': 1, 'Adam Rifkin': 1, 'Tamara
Jenkins': 1, 'Danny Perez': 1, 'Jamie Babbit': 1}), 'Bill Cobbs': defaultdict(<cl
ass 'int'>, {'James Bobin': 1, 'Michael Apted': 1, 'Tony Kaye': 1, 'Forest Whitak

```

```

er': 1, 'Mick Jackson': 1, 'Michael O. Sajbel': 1}}, 'Sydney Pollack': defaultdict(
t(<class 'int'>, {'Roger Michell': 1}), 'Adam Garcia': defaultdict(<class 'int'>,
{'David McNally': 1, 'Sara Sugarman': 1, 'Klaus Menzel': 1}), 'Lucille Ball': def
aultdict(<class 'int'>, {'Melville Shavelson': 1}), 'Dania Ramirez': defaultdict
(<class 'int'>, {'Joel Zwick': 1}), 'Demi  n Bichir': defaultdict(<class 'int'>,
{'Oliver Stone': 1, 'Chris Weitz': 1}), 'Richard Burgi': defaultdict(<class 'in
t'>, {'David R. Ellis': 1}), 'Kevin McNally': defaultdict(<class 'int'>, {'Peter
Howitt': 1, 'Irwin Winkler': 1}), 'Virginia Madsen': defaultdict(<class 'int'>,
{'David Lynch': 1, 'Alexander Payne': 1, 'Bobby Roth': 1}), 'Jim Sturgess': defau
ltdict(<class 'int'>, {'Julie Taymor': 1, 'Giuseppe Tornatore': 1, 'Kari Skoglan
d': 1}), 'Alexa PenaVega': defaultdict(<class 'int'>, {'John Gray': 1, 'Ulu Grosb
ard': 1, 'Rob Reiner': 1, 'Robert Rodriguez': 3, 'Darren Lynn Bousman': 1, 'Mark
Young': 1, 'Jon Gunn': 1}), 'John de Lancie': defaultdict(<class 'int'>, {'Harold
Ramis': 1, 'Marc Sch  lermann': 1}), 'Malese Jow': defaultdict(<class 'int'>, {'J
ohn Schultz': 1}), 'Costas Mandylor': defaultdict(<class 'int'>, {'Sean Penn': 1,
'Kevin Greutert': 2, 'Darren Lynn Bousman': 2}), 'Marcela Mar': defaultdict(<clas
s 'int'>, {'Mike Newell': 1}), 'Pierre Perrier': defaultdict(<class 'int'>, {'':
1}), 'Sam Waterston': defaultdict(<class 'int'>, {'George P. Cosmatos': 1}), 'Dan
iel Kaluuya': defaultdict(<class 'int'>, {'Oliver Parker': 1}), 'Clea DuVall': de
faultdict(<class 'int'>, {'Ben Affleck': 1, 'James Mangold': 1}), 'Mel Blanc': de
faultdict(<class 'int'>, {'Norman Ferguson': 1}), 'Judith Chapman': defaultdict(<
class 'int'>, {'Roger Kumble': 1}), 'Augustus Prew': defaultdict(<class 'int'>,
{'Burr Steers': 1}), 'Imelda Staunton': defaultdict(<class 'int'>, {'Peter Lord':
1, 'Ang Lee': 1}), 'John Witherspoon': defaultdict(<class 'int'>, {'Reginald Hudl
in': 1, 'Wes Craven': 1, 'Marcus Raboy': 1, 'Steve Carr': 1}), 'Debra Messing': d
efaultdict(<class 'int'>, {'Mark Pellington': 1, 'Edward Burns': 1}), 'Bono': def
aultdict(<class 'int'>, {'Larry Charles': 1, 'Catherine Owens': 1}), 'Robert Dav
i': defaultdict(<class 'int'>, {'John Glen': 1}), 'Fisher Stevens': defaultdict(<
class 'int'>, {'Julie Anne Robinson': 1, 'Fr  d  ric Auburtin': 1, 'Walter Hill':
1, 'Meiert Avis': 1, 'Avi Nesher': 1}), 'Tony Cox': defaultdict(<class 'int'>,
{'Sngmoo Lee': 1}), 'Omar Sy': defaultdict(<class 'int'>, {'Jean-Pierre Jeunet':
1}), 'Mekhi Phifer': defaultdict(<class 'int'>, {'Curtis Hanson': 1, 'Wayne Beac
h': 1, 'Preston A. Whitmore II': 1, 'Tim Blake Nelson': 1}), 'Jamel Debbouze': de
faultdict(<class 'int'>, {'Jamel Debbouze': 1}), 'Julian Sands': defaultdict(<cla
ss 'int'>, {'Gordon Chan': 1, 'Frank Marshall': 1, 'Steve Miner': 1, 'James Ivor
y': 1, 'Anthony Hickox': 1}), 'Robert Clohessy': defaultdict(<class 'int'>, {'Asg
er Leth': 1}), 'Joel McHale': defaultdict(<class 'int'>, {'David Frankel': 1}),
'Martin Kove': defaultdict(<class 'int'>, {'John G. Avildsen': 1, 'Mario Van Peeb
les': 1}), 'Tom Welling': defaultdict(<class 'int'>, {'Shawn Levy': 1}), 'Jerry F
errara': defaultdict(<class 'int'>, {'Peter Berg': 1, 'Michael Corrente': 1}), 'J
avier Botet': defaultdict(<class 'int'>, {'James Wan': 1, 'Andr  s Muschietti':
1, 'Brian Dorton': 1}), 'Andrew Garfield': defaultdict(<class 'int'>, {'David Fin
cher': 1, 'Terry Gilliam': 1, 'Mark Romanek': 1, 'Julian Jarrold': 1}), 'Carmen P
erez': defaultdict(<class 'int'>, {'Ken Kwapis': 1}), 'Beau Mirchoff': defaultdic
t(<class 'int'>, {'David Zucker': 1}), 'Kelly Rutherford': defaultdict(<class 'in
t'>, {'Wes Craven': 1}), 'Olivia Munn': defaultdict(<class 'int'>, {'Tim Story':
1, 'Scott Derrickson': 1, 'Douglas McGrath': 1}), 'Wes Studi': defaultdict(<class
'int'>, {'Michael Mann': 1}), 'Bokeem Woodbine': defaultdict(<class 'int'>, {'Tay
lor Hackford': 1, 'Kirk Wong': 1, 'Matthew Bright': 1, 'Christian Sesma': 1, 'Tom
my Oliver': 1}), 'Jason Alexander': defaultdict(<class 'int'>, {'Bobby Farrelly':
1}), 'Sally Kirkland': defaultdict(<class 'int'>, {'Oliver Stone': 1, 'Sharon Gre
ytak': 1}), 'Matthew Perry': defaultdict(<class 'int'>, {'Burr Steers': 1, 'Regin
ald Hudlin': 1, 'Damon Santostefano': 1}), 'Krista Allen': defaultdict(<class 'in
t'>, {'David R. Ellis': 1, 'John Gulager': 1}), 'Joaquim de Almeida': defaultdict
(<class 'int'>, {'John Moore': 1, 'James Manera': 1}), 'Pink': defaultdict(<class
'int'>, {'Nicholas Stoller': 1}), 'Lisa Ann Walter': defaultdict(<class 'int'>,
{'Peter Chelsom': 1, 'Steven Brill': 1}), 'Michael Jai White': defaultdict(<class
'int'>, {'Mark A.Z. Dipp  ': 1, 'Mic Rodgers': 1, 'Andrzej Bartkowiak': 1, 'Tyler
Perry': 2, 'Ekachai Uekrongtham': 1, 'Ernie Barbarash': 1, 'Kevin Carraway': 1,
'Michael Jai White': 1}), 'Rudy Youngblood': defaultdict(<class 'int'>, {'Mel Gib

```

```

son': 1, 'Russell Friedenberg': 1}}, 'Joe Don Baker': defaultdict(<class 'int'>,
{'John Glen': 1}), 'Topher Grace': defaultdict(<class 'int'>, {'NimrÃ³d Antal':
1, 'Robert Luketic': 1, 'Michael Dowse': 1, 'Jason Stone': 1}), 'Brian Dennehy':
defaultdict(<class 'int'>, {'Ivan Reitman': 1, 'Jean-FranÃ§ois Richet': 1, 'Rober
t Moresco': 1, 'Pete Jones': 1}), 'Gena Rowlands': defaultdict(<class 'int'>, {'I
ain Softley': 1}), 'Busy Philipps': defaultdict(<class 'int'>, {'Paul Weiland':
1, 'Keenen Ivory Wayans': 1, 'Joel Edgerton': 1}), 'Johnny Cannizzaro': defaulttdi
ct(<class 'int'>, {'Clint Eastwood': 1}), 'Lorraine Bracco': defaultdict(<class
'int'>, {'John McTiernan': 1}), 'Sam Trammell': defaultdict(<class 'int'>, {'Coli
n Strause': 1, 'Chuck Russell': 1}), 'Macaulay Culkin': defaultdict(<class 'in
t'>, {'Donald Petrie': 1, 'Chris Columbus': 2, 'Howard Zieff': 1, 'Brian Dannell
y': 1, 'Fenton Bailey': 1}), 'Bobby Lee': defaultdict(<class 'int'>, {'Greg Motto
la': 1}), 'Miriam Margolyes': defaultdict(<class 'int'>, {'Anne Fletcher': 1, 'Ch
ris Noonan': 1, 'IstvÃ¡n SzabÃ³': 1, 'Barbra Streisand': 1}), 'Alison Brie': defa
ultdict(<class 'int'>, {'Wes Craven': 1, 'Christian Ditter': 1, 'Nicholas Stolle
r': 1}), 'Michael Wincott': defaultdict(<class 'int'>, {'Oliver Stone': 2, 'Alex
Proyas': 1}), 'James Wilcox': defaultdict(<class 'int'>, {'Jake Kasdan': 1}), 'Em
ma Bell': defaultdict(<class 'int'>, {'Steven Quale': 1, 'Davis Guggenheim': 1}),
'Jimmy Fallon': defaultdict(<class 'int'>, {'Bobby Farrelly': 1, 'Dave Borthwic
k': 1}), 'Bruce Dinsmore': defaultdict(<class 'int'>, {'': 1}), 'Noah Bean': defa
ultdict(<class 'int'>, {'Roger Michell': 1}), '50 Cent': defaultdict(<class 'in
t'>, {'Jim Sheridan': 1}), 'Oliver Hudson': defaultdict(<class 'int'>, {'Sam Weis
man': 1, 'Glen Morgan': 1}), 'Catherine Deneuve': defaultdict(<class 'int'>, {'Pe
ter Hyams': 1, 'Lars von Trier': 1, 'AndrÃ© TÃ©chinÃ©': 1, 'FranÃ§ois Ozon': 1,
'Vincent Paronnaud': 1, 'Dick Richards': 1, 'FranÃ§ois Truffaut': 1}), 'Stephen R
ea': defaultdict(<class 'int'>, {'Stephen Hopkins': 1, 'Neil Jordan': 1, 'LluÃ\xad
s QuÃ\xadlez': 1}), 'Jordi MollÃ\xa0': defaultdict(<class 'int'>, {'Olivier Mega
ton': 1, 'Roland JoffÃ©': 1}), 'Mira Sorvino': defaultdict(<class 'int'>, {'Irwin
Winkler': 1, 'Antoine Fuqua': 1, 'Spike Lee': 1, 'Christopher M. Bessette': 1, 'G
riffin Dunne': 1}), 'Dane Cook': defaultdict(<class 'int'>, {'Joseph Kahn': 1, 'M
ark Helfrich': 1, 'Howard Deutch': 1, 'Greg Coolidge': 1}), 'Bobbie Phillips': de
faultdict(<class 'int'>, {'Paul Verhoeven': 1}), 'Carol Kane': defaultdict(<class
'int'>, {'Elaine May': 1}), 'Richard Epcar': defaultdict(<class 'int'>, {'John Ca
rpenter': 1}), 'Mathieu Kassovitz': defaultdict(<class 'int'>, {'Jean-Pierre Jeun
et': 1}), 'Mary-Kate Olsen': defaultdict(<class 'int'>, {'Dennie Gordon': 1, 'Jon
athan Levine': 1}), 'Omar Epps': defaultdict(<class 'int'>, {'Charles Shyer': 1,
'Charles S. Dutton': 1, 'Wes Craven': 1, 'Takeshi Kitano': 1, 'Rick Famuyiwa':
1}), 'Justin Theroux': defaultdict(<class 'int'>, {'Danny DeVito': 1, 'David Wai
n': 1, 'Greg Berlanti': 1}), 'M. Emmet Walsh': defaultdict(<class 'int'>, {'Jerry
Jameson': 1}), 'Bridgette Wilson-Sampras': defaultdict(<class 'int'>, {'Jean-Mari
e PoirÃ©': 1}), 'John Gatins': defaultdict(<class 'int'>, {'Brian Robbins': 1}),
'Sarah Roemer': defaultdict(<class 'int'>, {'D.J. Caruso': 1}), 'James Brolin': d
efaultdict(<class 'int'>, {'Richard Shepard': 1}), 'Jennifer Jason Leigh': defaul
tdict(<class 'int'>, {'Joel Coen': 1, 'John Maybury': 1, 'David Cronenberg': 1,
'Dan Curtis': 1, 'Jane Campion': 1, 'Tuck Tucker': 1, 'Duke Johnson': 1, 'Amy Hec
kerling': 1}), 'Jay Hernandez': defaultdict(<class 'int'>, {'Sergey Bodrov': 1,
'Boaz Yakin': 1, 'Neil LaBute': 1, 'Eli Roth': 2}), 'Tamannaah Bhatia': defaulttdi
ct(<class 'int'>, {'S.S. Rajamouli': 1}), 'Arliss Howard': defaultdict(<class 'in
t'>, {'Robert Schwentke': 1}), 'Colin Ferguson': defaultdict(<class 'int'>, {'Mic
hael Lehmann': 1}), 'Adam Baldwin': defaultdict(<class 'int'>, {'Joss Whedon': 1,
'Stephen Hopkins': 1, 'Richard Donner': 1, 'Robert Redford': 1}), 'Takeshi Kanesh
iro': defaultdict(<class 'int'>, {'John Woo': 1, 'Yimou Zhang': 1}), 'Ethan Embr
y': defaultdict(<class 'int'>, {'Andy Tennant': 1, 'Harry Elfont': 1, 'Danny Lein
er': 1, 'Tim McCanlies': 1}), 'Austin Pendleton': defaultdict(<class 'int'>, {'Jo
nathan Lynn': 1, 'Billy Wilder': 1}), 'Tony Chiu Wai Leung': defaultdict(<class
'int'>, {'Kar-Wai Wong': 1}), 'Blythe Auffarth': defaultdict(<class 'int'>, {'Yue
feng Song': 1}), 'Sarah Silverman': defaultdict(<class 'int'>, {'David Mirkin':
1, 'Bobby Farrelly': 1, 'Scott Alexander': 1}), 'D.B. Woodside': defaultdict(<cla
ss 'int'>, {'Andy Fickman': 1}), 'Sonia Braga': defaultdict(<class 'int'>, {'Luis
Mandoki': 1}), 'Beth Grant': defaultdict(<class 'int'>, {'Stephen Herek': 1, 'Jas

```

```

on Bateman': 1}}, "Mo'Nique": defaultdict(<class 'int'>, {'Lee Daniels': 1, 'Malcolm D. Lee': 1, 'John Singleton': 1, 'Jessy Terrero': 1, 'Mark Brown': 1, 'DJ Poo
h': 1, 'Nnegest LikkÃ': 1}}, 'Wilford Brimley': defaultdict(<class 'int'>, {'John Carpenter': 1, 'Peter Segal': 1, 'Richard Dutcher': 1}}, 'Gregory Scott Cummins': defaultdict(<class 'int'>, {'Jeb Stuart': 1}}, 'Tom Skerritt': defaultdict(<class 'int'>, {'Steve Miner': 1, 'Dominic Sena': 1, 'David Cronenberg': 1, 'Gary S
herman': 1, 'Ridley Scott': 1, 'Mario Van Peebles': 1}}, 'Mike Howard': defaultdict(<class 'int'>, {'Paul Thomas Anderson': 1}}, 'Hayden Christensen': defaultdict(<class 'int'>, {'David Leland': 1, 'Sarik Andreasyan': 1, 'Joby Harold': 1, 'George Hickenlooper': 1, 'Billy Ray': 1, 'Michael Polish': 1}}, 'Patton Oswalt': defaultdict(<class 'int'>, {'Paul Thomas Anderson': 1, 'Todd Strauss-Schulson': 1}}, 'Laura-Leigh': defaultdict(<class 'int'>, {'Rawson Marshall Thurber': 1}}, 'Yuliya Snigir': defaultdict(<class 'int'>, {'Fedor Bondarchuk': 1}}, 'Kathleen Quinlan': defaultdict(<class 'int'>, {'Jonathan Mostow': 1}}, 'Bernie Casey': defaultdict(<class 'int'>, {'Irvin Kershner': 1}}, 'Charlie McDermott': defaultdict(<class 'int'>, {'Steve Pink': 1, 'Courtney Hunt': 1}}, 'Rick Yune': defaultdict(<class 'int'>, {'Scott Hicks': 1, 'RZA': 1}}, 'Blair Underwood': defaultdict(<class 'int'>, {'Andrew Niccol': 1, 'Michael Schultz': 1}}, 'Stockard Channing': defaultdict(<class 'int'>, {'Andrew Bergman': 1, 'Gary David Goldberg': 1, 'Hugh Wilson': 1, 'Patrick Stettner': 1}}, 'Cheryl Hines': defaultdict(<class 'int'>, {'Kirk De Micco': 1}}, 'Mike Tyson': defaultdict(<class 'int'>, {'Wilson Yip': 1}}, 'Rene Russo': defaultdict(<class 'int'>, {'Richard Donner': 1}}, 'Catherine Dyer': defaultdict(<class 'int'>, {'John Lee Hancock': 1}}, 'Jon Foo': defaultdict(<class 'int'>, {'': 1, 'Prachya Pinkaew': 1}}, 'Amy Schumer': defaultdict(<class 'int'>, {'Judd Apatow': 1}}, 'Michael Angarano': defaultdict(<class 'int'>, {'Mike Mitchell': 1, 'Thomas Vinterberg': 1, 'Gavin Wiesen': 1, 'David Gordon Green': 1}}, 'Natascha Henstridge': defaultdict(<class 'int'>, {'Roger Donaldson': 1, 'Ringo Lam': 1, 'Richard Boddington': 1}}, 'Dylan Baker': defaultdict(<class 'int'>, {'Tarsem Singh': 1, 'Clare Kilner': 1, 'Bob Odenkirk': 1}}, 'Scott Glenn': defaultdict(<class 'int'>, {'Randall Wallace': 1, 'John McTiernan': 1, 'Lawrence Kasdan': 1, 'David Hackl': 1}}, 'Charlyne Yi': defaultdict(<class 'int'>, {'Judd Apatow': 1}}, 'Craig Parker': defaultdict(<class 'int'>, {'Patrick Tatopoulos': 1}}, 'Heather O'Rourke': defaultdict(<class 'int'>, {'Tobe Hooper': 1}}, 'Danneel Ackles': defaultdict(<class 'int'>, {'Alan Poul': 1, 'Jon Hurwitz': 1}}, 'Ashley Williams': defaultdict(<class 'int'>, {'Luke Greenfield': 1}}, 'Jodi Lyn O'Keefe': defaultdict(<class 'int'>, {'': 1}}, 'Tatum O'Neal': defaultdict(<class 'int'>, {'Michael Ritchie': 1}}, 'Emilia Fox': defaultdict(<class 'int'>, {'Roman Polanski': 1}}, 'Kisha Castle-Hughes': defaultdict(<class 'int'>, {'Catherine Hardwicke': 1, 'Niki Caro': 1}}, 'Robert Richardson': defaultdict(<class 'int'>, {'Jaume Collet-Serra': 1}}, 'Aaliyah': defaultdict(<class 'int'>, {'Michael Rymer': 1}}, 'Brian Blessed': defaultdict(<class 'int'>, {'Mike Hodges': 1, 'Kenneth Branagh': 1}}, 'Cicely Tyson': defaultdict(<class 'int'>, {'Rob Cohen': 1, 'Wayne Wang': 1, 'Bryan Barber': 1, 'Darren Grant': 1}}, 'Daniel Mays': defaultdict(<class 'int'>, {'Susanna White': 1}}, 'Frank Whaley': defaultdict(<class 'int'>, {'Danny DeVito': 1}}, 'Mitch Pileggi': defaultdict(<class 'int'>, {'Chris Carter': 1}}, 'Ken Arnold': defaultdict(<class 'int'>, {'John Singleton': 1}}, 'Drake Bell': defaultdict(<class 'int'>, {'Craig Mazin': 1, 'Stephen Frears': 1, 'Taedong Park': 1, 'Deb Hagan': 1}}, 'Alona Tal': defaultdict(<class 'int'>, {'Allen Hughes': 1}}, 'Shannon Elizabeth': defaultdict(<class 'int'>, {'Wes Craven': 1, 'Steve Beck': 1, 'Gregory Poirier': 1}}, 'Jim Gaffigan': defaultdict(<class 'int'>, {'Anne Fletcher': 1, 'George Ratliff': 1, 'Sol Tryon': 1}}, 'Alex Russell': defaultdict(<class 'int'>, {'Kimble Rendall': 1}}, 'Rosanna Arquette': defaultdict(<class 'int'>, {'John Milius': 1}}, 'Tim Meadows': defaultdict(<class 'int'>, {'Jake Kasdan': 1}}, 'Gregory Smith': defaultdict(<class 'int'>, {'Les Mayfield': 1, 'Bronwen Hughes': 1, 'Jason Eisener': 1}}, 'Michiel Huisman': defaultdict(<class 'int'>, {'Jean-Marc VallÃe': 2, 'Paul Verhoeven': 1}}, 'Paul Sorvino': defaultdict(<class 'int'>, {'Hark Tsui': 1, 'Warren Beatty': 1, 'Ãx89mile Gaudreault': 1, 'Darren Lynn Bousman': 1}}, 'Mireille Enos': defaultdict(<class 'int'>, {'David Ayer': 1}}, 'Kevin Zegers': defaultdict(<class 'int'>, {'Peter Hewitt': 1, 'Rob Schmidt': 1, 'Charles Martin Smith': 1, 'Derick Martini': 1, 'Duncan Tucker': 1}}, 'Marley Shelton': defau

```

```

ltdict(<class 'int'>, {'Ronny Yu': 1, 'Jamie Blanks': 1, 'David Caffrey': 1}), 'A
rchie Panjabi': defaultdict(<class 'int'>, {'Ridley Scott': 1, 'Fernando Meirelle
s': 1, 'Gurinder Chadha': 1, "Damien O'Donnell": 1}), 'Ruth Wilson': defaultdict
(<class 'int'>, {'': 1}), 'Oliver Reed': defaultdict(<class 'int'>, {'Moustapha A
kkad': 1, 'Carol Reed': 1, 'Guy Hamilton': 1}), 'Jeremy W. Auman': defaultdict(<c
lass 'int'>, {'Ang Lee': 1}), 'Maricel Ã\x81lvarez': defaultdict(<class 'int'>,
{'Alejandro G. IÃ±Ã©rritu': 1}), 'Richard E. Grant': defaultdict(<class 'int'>,
{'Martin Scorsese': 1, 'Uli Edel': 1, 'Richard Shepard': 1}), 'David Paymer': def
aultdict(<class 'int'>, {'Hugh Johnson': 1, 'Rob Reiner': 1}), 'Rumi Hiiragi': de
faultdict(<class 'int'>, {'Hayao Miyazaki': 1}), 'Tiago Riani': defaultdict(<clas
s 'int'>, {'George Tillman Jr.': 1}), 'Liam Aiken': defaultdict(<class 'int'>,
{'Hugh Hudson': 1, 'John Hoffman': 1, 'Nicholas Hytner': 1, 'Michael Winterbotto
m': 1}), 'Terry Pheto': defaultdict(<class 'int'>, {'Justin Chadwick': 1, 'Bille
August': 1, 'Gavin Hood': 1}), 'Nia Long': defaultdict(<class 'int'>, {'Raja Gosn
ell': 1, 'Rupert Wainwright': 1, 'Brian Levant': 1, 'Mennan Yapo': 1, 'Peter Aten
cio': 1, 'F. Gary Gray': 1}), 'Kyle Hebert': defaultdict(<class 'int'>, {'Jeremy
Degruson': 1}), 'Ryan Guzman': defaultdict(<class 'int'>, {'Scott Speer': 1, 'Rob
Cohen': 1, 'Rita Merson': 1}), 'Lin Shaye': defaultdict(<class 'int'>, {'David R.
Ellis': 1, 'Leigh Whannell': 1, 'James Wan': 2, 'Stiles White': 1, 'William Euban
k': 1}), 'Laura Harring': defaultdict(<class 'int'>, {'Jonathan Hensleigh': 1, 'G
len Morgan': 1}), 'Ryan Kruger': defaultdict(<class 'int'>, {'Neil Marshall':
1}), 'James Coburn': defaultdict(<class 'int'>, {'Brian Levant': 1, 'Sam Peckinp
ah': 2, 'John Sturges': 1}), 'Ana de la Reguera': defaultdict(<class 'int'>, {'Jar
ed Hess': 1}), 'Matt Lucas': defaultdict(<class 'int'>, {'Paul Feig': 1}), 'Brand
on T. Jackson': defaultdict(<class 'int'>, {'John Whitesell': 1, 'Erik White':
1}), 'Illeana Douglas': defaultdict(<class 'int'>, {'Frank Marshall': 1, 'David K
oepp': 1}), 'Callum Rennie': defaultdict(<class 'int'>, {'Jean-Pierre Jeunet': 1,
'Les Mayfield': 1, 'Christopher Nolan': 1}), 'Ken Howard': defaultdict(<class 'in
t'>, {'John Gatins': 1, 'Irwin Winkler': 1}), 'Frances Fisher': defaultdict(<clas
s 'int'>, {'Peter Howitt': 1, 'Vadim Perelman': 1, 'Catherine Hardwicke': 1, 'Kar
en Moncrieff': 1}), 'Alice Braga': defaultdict(<class 'int'>, {'Miguel Sapochni
k': 1, 'David Mamet': 1, 'Fernando Meirelles': 1}), 'George Kennedy': defaultdict
(<class 'int'>, {'Mark Dindal': 1, 'Peter Segal': 1, 'Rupert Wyatt': 1, 'David Zu
cker': 1, 'Sam Levinson': 1, 'Michael Gornick': 1}), 'Joan Plowright': defaultdic
t(<class 'int'>, {'Tom Reeve': 1}), 'Rob Riggle': defaultdict(<class 'int'>, {'Na
nette Burstein': 1, 'Luke Greenfield': 1, 'Neal Brennan': 1}), 'Alicia Witt': def
aultdict(<class 'int'>, {'Stephen Herek': 1, 'Tyler Perry': 1, 'Jamie Blanks': 1,
'Mike Binder': 1, 'John Waters': 1}), 'Desmond Llewelyn': defaultdict(<class 'in
t'>, {'Lewis Gilbert': 1, 'Terence Young': 1, 'Guy Hamilton': 1}), 'Sharlto Cople
y': defaultdict(<class 'int'>, {'Neill Blomkamp': 1}), 'John Doman': defaultdict
(<class 'int'>, {'Clint Eastwood': 1}), 'Emily Osment': defaultdict(<class 'in
t'>, {'Peter Chelsom': 1}), 'Elizabeth Daily': defaultdict(<class 'int'>, {'Stig
Bergqvist': 1, 'Igor Kovalyov': 1, 'John Eng': 1, '': 1}), 'George Carlin': defau
ltdict(<class 'int'>, {'Barbra Streisand': 1}), 'Hope Davis': defaultdict(<class
'int'>, {'Alexander Payne': 1, 'Lasse HallstrÃ©m': 1}), 'Cory Hardrict': defaultd
ict(<class 'int'>, {'Jonathan Levine': 1}), 'Zach Gilford': defaultdict(<class 'i
nt'>, {'': 2, 'Jee-woon Kim': 1, 'Matt Bettinelli-Olpin': 1}), 'Josh Charles': de
faultdict(<class 'int'>, {'John Singleton': 1, 'Tim Hill': 1}), 'Tina Fey': defau
ltdict(<class 'int'>, {'Michael McCullers': 1, 'Shawn Levy': 1, 'Mark Waters': 1,
'Paul Weitz': 1, 'Matt Maiellaro': 1}), 'Antony Starr': defaultdict(<class 'in
t'>, {'Steven Brill': 1}), 'Kurt Fuller': defaultdict(<class 'int'>, {'Woody Alle
n': 1}), 'Brenda Song': defaultdict(<class 'int'>, {'John Schultz': 1}), 'Patrick
Macnee': defaultdict(<class 'int'>, {'John Glen': 1, 'Joe Dante': 1}), 'Mark Gati
ss': defaultdict(<class 'int'>, {'Steve Box': 1, 'Jez Butterworth': 1}), 'Kate Up
ton': defaultdict(<class 'int'>, {'Bobby Farrelly': 1}), 'Mehcad Brooks': default
dict(<class 'int'>, {'James Gartner': 1, 'H.M. Coakley': 1}), 'Edgar Arreola': de
faultdict(<class 'int'>, {'Denis Villeneuve': 1}), 'Alyson Stoner': defaultdict(<
class 'int'>, {'Jon M. Chu': 1}), 'Faizon Love': defaultdict(<class 'int'>, {'Joh
n Stockwell': 1, 'Martin Lawrence': 1}), 'Danny Huston': defaultdict(<class 'in
t'>, {'David Slade': 1}), 'Carmen Electra': defaultdict(<class 'int'>, {'Jason Fr

```

iedberg': 2, 'David Zucker': 1}), 'Yaphet Kotto': defaultdict(<class 'int'>, {'Paul Michael Glaser': 1, 'Guy Hamilton': 1, 'Michael Roemer': 1}), 'Jessica Bardenn': defaultdict(<class 'int'>, {'Joe Wright': 1}), 'Brian Thompson': defaultdict(<class 'int'>, {'John R. Leonetti': 1}), 'Debi Mazar': defaultdict(<class 'int'>, {'': 1}), 'Devin Ratray': defaultdict(<class 'int'>, {'Martha Coolidge': 1, 'Alexander Payne': 1, 'Jeremy Saulnier': 1}), 'Craig Ferguson': defaultdict(<class 'int'>, {'Stephen J. Anderson': 1, 'Kevin Allen': 1}), 'Warren Beatty': defaultdict(<class 'int'>, {'Warren Beatty': 1}), 'Tika Sumpter': defaultdict(<class 'int'>, {'Tate Taylor': 1}), 'Rainn Wilson': defaultdict(<class 'int'>, {'Ivan Reitman': 1}), 'Sarah Polley': defaultdict(<class 'int'>, {'Vincenzo Natali': 1, 'Doug Liman': 1, 'Isabel Coixet': 1}), 'Eddie Cibrian': defaultdict(<class 'int'>, {'Bruce Hunt': 1, 'J.B. Rogers': 1, 'Tyler Perry': 1, 'Bill Duke': 1}), 'Jessica ParÁ': defaultdict(<class 'int'>, {'Paul McGuigan': 1}), 'Grace Phipps': defaultdict(<class 'int'>, {'Craig Gillespie': 1}), 'Saffron Burrows': defaultdict(<class 'int'>, {'Chris Roberts': 1}), 'Ian Whyte': defaultdict(<class 'int'>, {'James Wong': 1}), 'Annabella Sciorra': defaultdict(<class 'int'>, {'Andy Cadiff': 1}), 'Á\x93lafur Darri Á\x93lafsson': defaultdict(<class 'int'>, {'': 1}), 'Tyler Labine': defaultdict(<class 'int'>, {'Peter Howitt': 1}), 'Emir Kusturica': defaultdict(<class 'int'>, {'Neil Jordan': 1, 'Patrice Leconte': 1}), 'Emily Watson': defaultdict(<class 'int'>, {'Chris Noonan': 1, 'Alan Parker': 1, 'Brian Percival': 1, 'Richard E. Grant': 1}), 'Dong-gun Jang': defaultdict(<class 'int'>, {'Kaige Chen': 1}), 'Steve Howey': defaultdict(<class 'int'>, {'Corey Yuen': 1, 'Bo Zenga': 1}), 'Louise Bourgoin': defaultdict(<class 'int'>, {'Laurent Tirard': 1}), 'Paz Vega': defaultdict(<class 'int'>, {'Olivier Dahan': 1}), 'Konstantin Khabenskiy': defaultdict(<class 'int'>, {'Maksim Fadeev': 1, 'Timur Bekmambetov': 1, 'Aleksandr Veledinskiy': 1}), 'Sandra Ellis Lafferty': defaultdict(<class 'int'>, {'James Mangold': 1, 'Ryan Smith': 1}), 'Amanda Schull': defaultdict(<class 'int'>, {'': 1, 'Nicholas Hytner': 1}), 'Lisa Edelstein': defaultdict(<class 'int'>, {'Edward Norton': 1}), 'Dick Clark': defaultdict(<class 'int'>, {'George Clooney': 1}), 'Aldis Hodge': defaultdict(<class 'int'>, {'F. Gary Gray': 1}), 'Joe Lo Truglio': defaultdict(<class 'int'>, {'David Wain': 1, 'Steve Pink': 1, 'Frank Sebastiano': 1, 'Matt Walsh': 1}), 'Eddie Cahill': defaultdict(<class 'int'>, {'Gavin O'Connor': 1}), 'Shun Oguri': defaultdict(<class 'int'>, {'Shinji Aramaki': 1}), 'Julian Glover': defaultdict(<class 'int'>, {'John Glen': 1, 'Dominic Burns': 1}), 'Lilli Lavigne': defaultdict(<class 'int'>, {'Peter Chelsom': 1}), 'Mia Sara': defaultdict(<class 'int'>, {'Peter Hyams': 1}), 'David Lyons': defaultdict(<class 'int'>, {'Lasse HallstrÁm': 1}), 'Vinessa Shaw': defaultdict(<class 'int'>, {'Kenny Ortega': 1}), 'ZoÁ Kravitz': defaultdict(<class 'int'>, {'Scott Hicks': 1, 'Anna Boden': 1, 'Analeine Cal y Mayor': 1}), 'Elizabeth McGovern': defaultdict(<class 'int'>, {'Matthew Vaughn': 1}), 'Talulah Riley': defaultdict(<class 'int'>, {'Joe Wright': 1}), 'Philip Baker Hall': defaultdict(<class 'int'>, {'Janusz Kaminski': 1, 'John Dahl': 1}), 'Jacques Perrin': defaultdict(<class 'int'>, {'Jacques Perrin': 1}), 'Samaire Armstrong': defaultdict(<class 'int'>, {'Donald Petrie': 1}), 'Scott Grimes': defaultdict(<class 'int'>, {'Jay Roach': 1}), 'Mara Wilson': defaultdict(<class 'int'>, {'Michael Ritchie': 1, 'Britt Allcroft': 1}), 'Dominic Flores': defaultdict(<class 'int'>, {'David Gordon Green': 1}), 'Matt Smith': defaultdict(<class 'int'>, {'Burr Steers': 1}), 'Tenzin Thuthob Tsarong': defaultdict(<class 'int'>, {'Martin Scorsese': 1}), 'Louis Jourdan': defaultdict(<class 'int'>, {'John Glen': 1}), 'Harold Ramis': defaultdict(<class 'int'>, {'Judd Apatow': 1}), 'Richard Wilson': defaultdict(<class 'int'>, {'David Lean': 1}), 'Phil Davis': defaultdict(<class 'int'>, {'Richard Eyre': 1}), 'Ellen Burstyn': defaultdict(<class 'int'>, {'Callie Khouri': 1, 'William Friedkin': 1, 'Tim Chambers': 1, 'Darren Aronofsky': 1}), 'John Belushi': defaultdict(<class 'int'>, {'John Landis': 2}), 'America Ferrera': defaultdict(<class 'int'>, {'Sanaa Hamri': 1, 'Rick Famuyiwa': 1, 'Patricia Cardoso': 1, 'Patricia Riggan': 1}), 'Dolly Parton': defaultdict(<class 'int'>, {'Todd Graff': 1, 'Colin Higgins': 1}), 'Sharon Small': defaultdict(<class 'int'>, {'Chris Weitz': 1}), 'Gerald McRaney': defaultdict(<class 'int'>, {'Wolfgang Petersen': 1}), 'Jackie Gleason': defaultdict(<class 'int'>, {'': 1}), 'Piper Laurie': defaultdict(<class 'int'>, {'Walter Murch': 1}), 'Harry Lennix': defaultdict(<class 'int'>, {'E. Elias Merhige': 1, 'Kevin Rodney Sullivan': 1})

36/78



```

e': 1}}, 'Dany Boon': defaultdict(<class 'int'>, {'Pascal Chaumeil': 1, 'Dany Boon': 1}), 'Tom Berenger': defaultdict(<class 'int'>, {'Ron Maxwell': 1, 'Lance Hool': 1}), 'John Doe': defaultdict(<class 'int'>, {'Jonathan Kaplan': 1}), 'Kyra Sedgwick': defaultdict(<class 'int'>, {'Ole Bornedal': 1, 'Sidney Lumet': 1}), 'Alexander Skarsgård': defaultdict(<class 'int'>, {'Rod Lurie': 1, 'Lars von Trier': 1, 'Zal Batmanglij': 1, 'Marielle Heller': 1}), 'Brian Bosworth': defaultdict(<class 'int'>, {'Craig R. Baxley': 1}), 'Tanya Roberts': defaultdict(<class 'int'>, {'John Guillermin': 1}), 'Cheech Marin': defaultdict(<class 'int'>, {'Marcos Siega': 1, 'José Luis Valenzuela': 1}), 'Donald Faison': defaultdict(<class 'int'>, {'Jeffrey W. Byrd': 1, 'Amy Heckerling': 1, 'Bille Woodruff': 1, 'Scott Foley': 1, 'David Raynr': 1}), 'Don McKellar': defaultdict(<class 'int'>, {'Fernando Meirelles': 1}), 'Dallas Barnett': defaultdict(<class 'int'>, {'Andrew Adamson': 1}), 'Zoe Kazan': defaultdict(<class 'int'>, {'Richard Linklater': 1, 'Bradley Rust Gray': 1}), 'François Cluzet': defaultdict(<class 'int'>, {'Guillaume Canet': 1, 'Thomas Lilti': 1}), 'Scout Taylor-Compton': defaultdict(<class 'int'>, {'Taylor Hackford': 1, 'Rob Zombie': 1}), 'Yolande Moreau': defaultdict(<class 'int'>, {'Jean-Marie Deschamps': 1}), 'Sergey Garmash': defaultdict(<class 'int'>, {'Inna Evlannikova': 1}), 'Rod Steiger': defaultdict(<class 'int'>, {'Sergey Bondarchuk': 1}), 'Siu-Wong Fan': defaultdict(<class 'int'>, {'Teddy Chan': 1}), 'Minnie Driver': defaultdict(<class 'int'>, {'Bonnie Hunt': 1, 'Trey Parker': 1, 'Hayao Miyazaki': 1, 'George Armitage': 1, 'Oliver Parker': 1, 'Mel Smith': 1, 'Gina Prince-Bythewood': 1}), 'Gerry Bednob': defaultdict(<class 'int'>, {'Kevin Smith': 1}), 'Lena Horne': defaultdict(<class 'int'>, {'Sidney Lumet': 1}), 'Willie Garson': defaultdict(<class 'int'>, {'Ron Shelton': 1, 'Spike Jonze': 1}), 'Donald Pleasence': defaultdict(<class 'int'>, {'Richard Williams': 1, 'Lewis Gilbert': 1, 'John Carpenter': 1, 'Dominique Othenin-Girard': 1, 'Joe Chappelle': 1, 'Dwight H. Little': 1}), 'Seymour Cassel': defaultdict(<class 'int'>, {'Michael Dinner': 1, 'Randal Kleiser': 1}), 'Judith Barsi': defaultdict(<class 'int'>, {'Joseph Sargent': 1, 'Don Bluth': 1}), 'Tasha Smith': defaultdict(<class 'int'>, {'Fred Durst': 1}), 'Alessandro Nivola': defaultdict(<class 'int'>, {'Anne Fontaine': 1}), 'Tom Burke': defaultdict(<class 'int'>, {'Stephen Frears': 1}), 'Cole Hauser': defaultdict(<class 'int'>, {'': 1, 'Paul Abascal': 1}), 'Romola Garai': defaultdict(<class 'int'>, {'Mira Nair': 1, 'Christian Volckman': 1, 'Ruairi Robinson': 1, '': 1}), 'Bingbing Fan': defaultdict(<class 'int'>, {'Teddy Chan': 1, 'Tung-Shing Ye': 1}), 'Drew Powell': defaultdict(<class 'int'>, {'Mikael Häfström': 1}), 'Joan Rivers': defaultdict(<class 'int'>, {'Mel Brooks': 1}), 'Cem Yilmaz': defaultdict(<class 'int'>, {'Russell Crowe': 1}), 'Manish Dayal': defaultdict(<class 'int'>, {'Lasse Hallström': 1}), 'Roma Downey': defaultdict(<class 'int'>, {'Christopher Spencer': 1}), 'Denis O'Hare': defaultdict(<class 'int'>, {'Mikael Häfström': 1}), 'Carlos Ponce': defaultdict(<class 'int'>, {'Mike Bigelow': 1, 'Linda Mendonça': 1}), 'Jack Reynor': defaultdict(<class 'int'>, {'Ken Scott': 1}), 'Radivoje Bukvic': defaultdict(<class 'int'>, {'Susanna White': 1}), 'Amanda Detmer': defaultdict(<class 'int'>, {'Dennis Dugan': 1}), 'Zachary Gordon': defaultdict(<class 'int'>, {'David Bowers': 2, 'Rian Johnson': 1}), 'Rock Hudson': defaultdict(<class 'int'>, {'Blake Edwards': 1}), 'Ed Skrein': defaultdict(<class 'int'>, {'Camille Delamarre': 1}), 'Gary Lewis': defaultdict(<class 'int'>, {'Christian Carion': 1}), 'Romain Duris': defaultdict(<class 'int'>, {'Laurent Tirard': 1, 'Cécile Klapisch': 1}), 'Pablo Rago': defaultdict(<class 'int'>, {'Juan José Campanella': 1}), 'Ornella Muti': defaultdict(<class 'int'>, {'Woody Allen': 1}), 'Paige Turco': defaultdict(<class 'int'>, {'Stuart Gillard': 1}), 'Hunter Parrish': defaultdict(<class 'int'>, {'Richard LaGravenese': 1, 'David Jacobson': 1}), 'Omari Hardwick': defaultdict(<class 'int'>, {'Tyler Perry': 1, 'Salim Akil': 1, 'Benny Boom': 1, 'Ava DuVernay': 1}), 'Jason London': defaultdict(<class 'int'>, {'Katt Shear': 1}), 'Chris Harrison': defaultdict(<class 'int'>, {'': 1}), 'Elaine Hendrix': defaultdict(<class 'int'>, {'John Duigan': 1, 'Jim Issa': 1}), 'Bob Balaban': defaultdict(<class 'int'>, {'Steven Spielberg': 1}), 'Ian Somerhalder': defaultdict(<class 'int'>, {'Jim Sonzero': 1, 'Irwin Winkler': 1, 'Roger Avary': 1}), 'Dean Stockwell': defaultdict(<class 'int'>, {'Tony Scott': 1, 'Gregor Jordan': 1, 'Elia Kazan': 1}), 'Angus T. Jones': defaultdict(<class 'int'>, {'Adam Shankman': 1}), 'Bam Margera': defaultdict(<class 'int'>, {'Jeff Tremaine': 3}), 'Mackenzie

```

```

Foy': defaultdict(<class 'int'>, {'James Wan': 1, 'Stéphane Aubier': 1}), 'Miranda Cosgrove': defaultdict(<class 'int'>, {'Richard Linklater': 1, 'Scott Marshall': 1}), 'Shailene Woodley': defaultdict(<class 'int'>, {'Alexander Payne': 1, 'Josh Boone': 1, 'James Ponsoldt': 1}), 'Leonard Roberts': defaultdict(<class 'int'>, {'Charles Stone III': 1, 'Theodore Witcher': 1}), 'Sam Claflin': defaultdict(<class 'int'>, {'Thea Sharrock': 1}), 'John Amos': defaultdict(<class 'int'>, {'Tyler Perry': 1, 'Dario Argento': 1, 'Melvin Van Peebles': 1}), 'Barry Watson': defaultdict(<class 'int'>, {'Stephen Kay': 1}), 'Kath Soucie': defaultdict(<class 'int'>, {'Jun Falkenstein': 1, 'Frank Nissen': 1}), 'Lamman Rucker': defaultdict(<class 'int'>, {'': 1}), 'Billy Crudup': defaultdict(<class 'int'>, {'Tom McCarthy': 1, 'Bart Freundlich': 1, 'William H. Macy': 1, 'Alison Maclean': 1}), 'Richard Lawson': defaultdict(<class 'int'>, {'Kevin Rodney Sullivan': 1}), 'Violante Placido': defaultdict(<class 'int'>, {'Anton Corbijn': 1}), 'Jon Seda': defaultdict(<class 'int'>, {'Gregory Nava': 1}), 'Diedrich Bader': defaultdict(<class 'int'>, {'Jason Friedberg': 1, 'Chuck Sheetz': 1}), 'Jeremy Shada': defaultdict(<class 'int'>, {'Trey Parker': 1}), 'Gunnar Hansen': defaultdict(<class 'int'>, {'John Luessenhop': 1, 'Tobe Hooper': 1}), 'William McNamara': defaultdict(<class 'int'>, {'Jon Amiel': 1, 'Tanner Beard': 1, 'Jay Alaimo': 1, 'David DeCoteau': 1}), 'Marisa Saks': defaultdict(<class 'int'>, {'Malcolm D. Lee': 1}), 'Peter Firth': defaultdict(<class 'int'>, {'Kevin Reynolds': 1}), 'Isaiah Washington': defaultdict(<class 'int'>, {'Steve Beck': 1}), 'Sela Ward': defaultdict(<class 'int'>, {'Nelson McCormick': 1}), 'Johnny Lewis': defaultdict(<class 'int'>, {'Eric Valette': 1}), 'Kay Panabaker': defaultdict(<class 'int'>, {'Andrew Fleming': 1}), 'Jay Laga'aia': defaultdict(<class 'int'>, {'Michael Spierig': 1}), 'Leighton Meester': defaultdict(<class 'int'>, {'Thomas Bezucha': 1, 'Christian E. Christiansen': 1, 'Shana Feste': 1}), 'Kevin Sussman': defaultdict(<class 'int'>, {'Anthony Bell': 1}), 'Kurtwood Smith': defaultdict(<class 'int'>, {'Gus Van Sant': 1, 'Miguel Arteta': 1, 'Steve James': 1}), 'Samira Wiley': defaultdict(<class 'int'>, {'Henry Joost': 1}), 'Sheryl Lee': defaultdict(<class 'int'>, {'John Carpenter': 1}), 'Pablo Sevilla': defaultdict(<class 'int'>, {'': 1}), 'Janet Leigh': defaultdict(<class 'int'>, {'Alfred Hitchcock': 1, 'Jack Smight': 1}), 'Mako': defaultdict(<class 'int'>, {'Steve Rash': 1, 'Richard Fleischer': 1}), 'Shengyi Huang': defaultdict(<class 'int'>, {'Stephen Chow': 1}), 'Leah Remini': defaultdict(<class 'int'>, {'': 1}), 'Mustafa Haidari': defaultdict(<class 'int'>, {'Marc Forster': 1}), 'Thomas Lennon': defaultdict(<class 'int'>, {'Chris Koch': 1}), 'Andy Dick': defaultdict(<class 'int'>, {'Amy Heckerling': 1}), 'Ronald Reagan': defaultdict(<class 'int'>, {'Michael Moore': 1}), 'Andrew Fiscella': defaultdict(<class 'int'>, {'Nimród Antal': 1, 'John Erick Dowdle': 1}), 'James Callis': defaultdict(<class 'int'>, {'Michael O. Sajbel': 1}), 'Aasif Mandvi': defaultdict(<class 'int'>, {'David Koepf': 1}), 'Taylor Cole': defaultdict(<class 'int'>, {'Renny Harlin': 1}), 'Quinton Aaron': defaultdict(<class 'int'>, {'Michel Gondry': 1}), 'Aisha Tyler': defaultdict(<class 'int'>, {'James Wan': 1}), 'Scott Baio': defaultdict(<class 'int'>, {'Bob Clark': 1}), 'Debbie Reynolds': defaultdict(<class 'int'>, {'Michael Lembeck': 1}), 'Sammy Davis Jr.': defaultdict(<class 'int'>, {'Bob Fosse': 1}), 'Martin Dew': defaultdict(<class 'int'>, {'Paul Thomas Anderson': 1}), 'Chris Brown': defaultdict(<class 'int'>, {'Benson Lee': 1, 'Sylvain White': 1, 'Tim Story': 1}), 'Chriss Anglin': defaultdict(<class 'int'>, {'David Zucker': 1}), 'Kevin Alejandro': defaultdict(<class 'int'>, {'Fred Wolf': 1}), 'Catherine Lough Haggquist': defaultdict(<class 'int'>, {'Uwe Boll': 1}), 'Cameron Bright': defaultdict(<class 'int'>, {'Jonathan Glazer': 1}), 'Landon Liboiron': defaultdict(<class 'int'>, {'Paul Gross': 1}), 'Dylan Authors': defaultdict(<class 'int'>, {'Marc Abraham': 1}), 'Isabelle Adjani': defaultdict(<class 'int'>, {'Jean-Paul Rappeneau': 1}), 'Steve Guttenberg': defaultdict(<class 'int'>, {'Nancy Walker': 1, 'Barry Levinson': 1, 'Hugh Wilson': 1}), 'Donny Boaz': defaultdict(<class 'int'>, {'Angelo Pizzo': 1}), 'Damon Wayans': defaultdict(<class 'int'>, {'Richard Benjamin': 1}), 'William Hootkins': defaultdict(<class 'int'>, {'Katsuhiro Ōtomo': 1}), 'Esai Morales': defaultdict(<class 'int'>, {'Kevin Reynolds': 1, 'Luis Valdez': 1, 'Kristin Rizzo': 1}), 'Jiao Xu': defaultdict(<class 'int'>, {'Stephen Chow': 1}), 'Christian Clavier': defaultdict(<class 'int'>, {'Jean-Marie Poiré': 2}), 'Marco St. John': defaultdict(<class 'int'>, {'Kevin Munroe': 1}), 'Peter Va

```

```

ughan': defaultdict(<class 'int'>, {'': 1}), 'Necar Zadegan': defaultdict(<class
'int'>, {'Jane Clark': 1}), 'Andy Lau': defaultdict(<class 'int'>, {'Daniel Lee':
1, 'Alan Yuen': 1, 'Chia-Liang Liu': 1}), 'Mani Bharathi': defaultdict(<class 'in
t'>, {'Gnana Rajasekaran': 1}), 'Joey D. Vieira': defaultdict(<class 'int'>, {'To
m Walsh': 1}), 'Moritz Bleibtreu': defaultdict(<class 'int'>, {'Uli Edel': 1, 'To
m Tykwer': 1}), 'Mary McDonnell': defaultdict(<class 'int'>, {'Kevin Costner':
1}), 'Dan Byrd': defaultdict(<class 'int'>, {'Mark Rosman': 1, 'Alexandre Aja':
1}), 'David Arquette': defaultdict(<class 'int'>, {'Wes Craven': 1, 'Finn Taylo
r': 1}), 'Scatman Crothers': defaultdict(<class 'int'>, {'Stanley Kubrick': 1, 'M
ilos Forman': 1}), 'Jeffrey Combs': defaultdict(<class 'int'>, {'William Malone':
1}), 'Eric Mendenhall': defaultdict(<class 'int'>, {'Tyler Perry': 1}), 'Kari J.
Kramer': defaultdict(<class 'int'>, {'Rob Hawk': 1}), 'Taran Killam': defaultdict
(<class 'int'>, {'Shawn Levy': 1, 'Jerome Elston Scott': 1}), 'Mem Ferda': defaul
tdict(<class 'int'>, {'Lee Tamahori': 1, 'Julian Gilbey': 1}), "Nicole 'Snooki' P
olizzi": defaultdict(<class 'int'>, {'': 1}), 'Bunta Sugawara': defaultdict(<clas
s 'int'>, {'Hayao Miyazaki': 1}), 'Katrina Bowden': defaultdict(<class 'int'>,
{'Sean Anders': 1, 'Eli Craig': 1, 'Malcolm Goodwin': 1}), 'Darci Kistler': defau
ltdict(<class 'int'>, {'Emile Ardolino': 1}), 'Tony Nappo': defaultdict(<class 'i
nt'>, {'George A. Romero': 1}), 'Louis Koo': defaultdict(<class 'int'>, {'Danny P
ang': 1}), 'Craig Roberts': defaultdict(<class 'int'>, {'Nicholas Stoller': 1, 'R
ichard Ayoade': 1}), 'Luenell': defaultdict(<class 'int'>, {'Larry Charles': 1}),
'Linda Cardellini': defaultdict(<class 'int'>, {'Robert Luketic': 1, 'Nicholaus G
oossen': 1, 'James Gunn': 1}), 'Craig T. Nelson': defaultdict(<class 'int'>, {'Th
omas Bezucha': 1, 'Jeff Kanew': 1, '': 1}), 'Nia Vardalos': defaultdict(<class 'i
nt'>, {'Kirk Jones': 1, 'Donald Petrie': 1, 'Joel Zwick': 1}), 'Shane Black': def
aultdict(<class 'int'>, {'John McTiernan': 1}), 'Jeffrey Jones': defaultdict(<cla
ss 'int'>, {'Milos Forman': 1, 'Don Michael Paul': 1}), 'Raoul Bova': defaultdict
(<class 'int'>, {'Audrey Wells': 1}), 'Kristin Scott Thomas': defaultdict(<class
'int'>, {'Robert Altman': 1, 'Gilles Paquet-Brenner': 1, 'Mike Newell': 1, 'Sam T
aylor-Johnson': 1}), 'Peter Gallagher': defaultdict(<class 'int'>, {'': 1, 'Max M
ayer': 1, 'Steven Soderbergh': 1}), 'Sally Hawkins': defaultdict(<class 'int'>,
{'Woody Allen': 1, 'Phil Morrison': 1}), 'Christian Clemenson': defaultdict(<clas
s 'int'>, {'Paul Greengrass': 1}), 'Christian Monzon': defaultdict(<class 'int'>,
{'Bille Woodruff': 1}), 'Hulk Hogan': defaultdict(<class 'int'>, {'Rick Friedber
g': 1}), 'Bob Newhart': defaultdict(<class 'int'>, {'Mike Nichols': 1}), 'Collett
e Wolfe': defaultdict(<class 'int'>, {'Jody Hill': 1}), 'Nicholas Rowe': defaultd
ict(<class 'int'>, {'Barry Levinson': 1}), 'Kelsey Grammer': defaultdict(<class
'int'>, {'Kevin Tanchoen': 1}), 'Alan Ruck': defaultdict(<class 'int'>, {'Chris
Columbus': 1, 'Daniel Hsia': 1}), 'Jason Robards': defaultdict(<class 'int'>, {'W
illiam A. Fraker': 1}), 'Nancy Allen': defaultdict(<class 'int'>, {'Brian De Palm
a': 1}), 'Rebecca Budig': defaultdict(<class 'int'>, {'Courtney Solomon': 1}), 'J
oan Allen': defaultdict(<class 'int'>, {'Ang Lee': 1, 'Lenny Abrahamson': 1}), 'Y
aya DaCosta': defaultdict(<class 'int'>, {'Rob Reiner': 1}), 'David Margulies': d
efaultdict(<class 'int'>, {'Adrian Lyne': 1}), 'Ian McDiarmid': defaultdict(<clas
s 'int'>, {'Matthew Robbins': 1}), 'David Masterson': defaultdict(<class 'int'>,
{'Michael Hoffman': 1}), 'Tadanobu Asano': defaultdict(<class 'int'>, {'Sergey Bo
drov': 1}), 'Barry Bostwick': defaultdict(<class 'int'>, {'Hal Needham': 1}), 'Ju
lie Christie': defaultdict(<class 'int'>, {'Kenneth Branagh': 1, 'David Lean':
1}), 'Poppy Montgomery': defaultdict(<class 'int'>, {'': 1}), 'Ulrich Tukur': def
aultdict(<class 'int'>, {'Michael Haneke': 1}), 'James Hetfield': defaultdict(<cl
ass 'int'>, {'NimrÃ³d Antal': 1}), 'Austin Nichols': defaultdict(<class 'int'>,
{'Gregor Jordan': 1}), 'Edgar RamÃ\xadrez': defaultdict(<class 'int'>, {'': 1}),
'Takuya Kimura': defaultdict(<class 'int'>, {'Tran Anh Hung': 1, 'Takashi Yamazak
i': 1}), 'Greg Nicotero': defaultdict(<class 'int'>, {'George A. Romero': 1}), 'E
rnie Reyes Jr.': defaultdict(<class 'int'>, {'Richard Fleischer': 1}), 'Vanessa F
erlito': defaultdict(<class 'int'>, {'Tyler Perry': 1}), 'Jason Momoa': defaulttdi
ct(<class 'int'>, {'David Hayter': 1}), 'Cassie Ventura': defaultdict(<class 'in
t'>, {'Jon M. Chu': 1}), 'Henry Ian Cusick': defaultdict(<class 'int'>, {'Xavier
Gens': 1, 'Philip Saville': 1, 'Stephen Herek': 1}), 'Jennifer Hudson': defaulttdi
ct(<class 'int'>, {'Kasi Lemmons': 1, 'Darrell Roodt': 1}), 'Kirk Fox': defaulttdi

```

```

ct(<class 'int'>, {'Matt Dillon': 1}), 'Fionnula Flanagan': defaultdict(<class 'i
nt'>, {'Alejandro Amen  bar': 1}), 'Jeremy Brett': defaultdict(<class 'int'>, {'G
eorge Cukor': 1}), 'Chris Moss': defaultdict(<class 'int'>, {'Jay Chandrasekhar':
1}), 'Emmanuelle Vaugier': defaultdict(<class 'int'>, {'Michael Lehmann': 1, 'Dar
ren Lynn Bousman': 1}), 'Kevin Grevioux': defaultdict(<class 'int'>, {'Sam Weisma
n': 1}), 'Morgan Saylor': defaultdict(<class 'int'>, {'Niki Caro': 1}), 'T.I.': d
efaultdict(<class 'int'>, {'Chris Robinson': 1}), 'Thomas F. Wilson': defaultdict
(<class 'int'>, {'Trent Cooper': 1, 'Fred Walton': 1}), 'Ava Gardner': defaultdic
t(<class 'int'>, {'Nicholas Ray': 1}), 'Lou Taylor Pucci': defaultdict(<class 'in
t'>, {'Fede Alvarez': 1}), 'DeRay Davis': defaultdict(<class 'int'>, {'Ian Iqbal
Rashid': 1}), 'Aidan Quinn': defaultdict(<class 'int'>, {'Rowdy Herrington': 1,
'Rick Bieber': 1, 'Conor McPherson': 1, 'Maggie Greenwald': 1, 'Mary Pat Kelly':
1}), 'Tippi Hedren': defaultdict(<class 'int'>, {'Noel Marshall': 1}), 'Brenda Fr
icker': defaultdict(<class 'int'>, {'Joel Schumacher': 1}), 'Julianna Guill': def
aultdict(<class 'int'>, {'Todd Lincoln': 1}), 'Hal Holbrook': defaultdict(<class
'int'>, {'Oliver Stone': 1}), 'Danny Strong': defaultdict(<class 'int'>, {'Joe Nu
ssbaum': 1}), 'Bruno Kirby': defaultdict(<class 'int'>, {'Rob Reiner': 1}), 'Etha
n Suplee': defaultdict(<class 'int'>, {'Todd Phillips': 1, 'Tony Kaye': 1, 'Kevin
Smith': 2, 'Morgan J. Freeman': 1}), 'B  r  nice Bejo': defaultdict(<class 'in
t'>, {'Michel Hazanavicius': 1}), 'Clive Russell': defaultdict(<class 'int'>, {'C
yrus Nowrasteh': 1}), 'Jennifer Esposito': defaultdict(<class 'int'>, {'Perry And
elin Blake': 1, 'Daniel Taplitz': 1}), 'Ivana Baquero': defaultdict(<class 'in
t'>, {'Guillermo del Toro': 1}), 'Leslie Bibb': defaultdict(<class 'int'>, {'John
Whitesell': 1, 'Sam Miller': 1, 'Goran Dukic': 1, 'Catherine Jelski': 1}), 'Britt
any Daniel': defaultdict(<class 'int'>, {'Dennie Gordon': 1, 'Jay Chandrasekhar':
1}), 'Bolo Yeung': defaultdict(<class 'int'>, {'Sheldon Lettich': 1, 'Newt Arnol
d': 1}), 'Bill Bailey': defaultdict(<class 'int'>, {'Edgar Wright': 1}), 'Gustaf
Skarsg  rd': defaultdict(<class 'int'>, {'Peter Flinth': 1}), 'Gretchen Mol': def
aultdict(<class 'int'>, {'Josef Rusnak': 1, 'Jason Alexander': 1}), 'Farrah Fawce
tt': defaultdict(<class 'int'>, {'Lance Rivera': 1, 'Michael Anderson': 1}), 'Loc
hlyn Munro': defaultdict(<class 'int'>, {'Bruce Paltrow': 1, 'Fred Savage': 1}),
'Alison Lohman': defaultdict(<class 'int'>, {'Susanne Bier': 1, 'Michael Mayer':
1}), 'Charles Napier': defaultdict(<class 'int'>, {'Kenneth Johnson': 1, 'Russ Me
yer': 1}), 'Paz de la Huerta': defaultdict(<class 'int'>, {'Gaspar No  ': 1}), 'S
teven Van Zandt': defaultdict(<class 'int'>, {'': 1}), 'David Alpay': defaultdict
(<class 'int'>, {'Atom Egoyan': 1}), 'Bruce Dern': defaultdict(<class 'int'>, {'W
illiam Bindley': 1, 'Joe Dante': 1, 'Douglas Trumbull': 1}), 'Carole Bouquet': de
faultdict(<class 'int'>, {'G  rard Krawczyk': 1}), 'Dustin Milligan': defaultdict
(<class 'int'>, {'James Gunn': 1}), 'Chen Chang': defaultdict(<class 'int'>, {'An
g Lee': 1}), 'Bex Taylor-Klaus': defaultdict(<class 'int'>, {'': 1}), 'Nicole Dio
nne': defaultdict(<class 'int'>, {'JK Youn': 1}), "Beverly D'Angelo": defaultdict
(<class 'int'>, {'Harold Ramis': 1, 'John Turturro': 1}), 'Jackson Nicoll': defau
ltdict(<class 'int'>, {'Jeff Tremaine': 1}), 'Roger Allam': defaultdict(<class 'i
nt'>, {'Stephen Frears': 1}), 'Frankie Muniz': defaultdict(<class 'int'>, {'Shawn
Levy': 1}), 'Clayton Snyder': defaultdict(<class 'int'>, {'Jim Fall': 1}), 'Jorda
na Brewster': defaultdict(<class 'int'>, {'Robert Rodriguez': 1, 'Angela Robinso
n': 1}), "Rae'Ven Kelly": defaultdict(<class 'int'>, {'Brian Gibson': 1}), 'Claud
e Rains': defaultdict(<class 'int'>, {'David Lean': 1, 'Frank Capra': 1}), 'Tony
Goldwyn': defaultdict(<class 'int'>, {'Dennis Iliadis': 1, 'Tom McLoughlin': 1}),
'Jack Davenport': defaultdict(<class 'int'>, {'Clare Kilner': 1}), 'Jessie T. Ush
er': defaultdict(<class 'int'>, {'Thomas Carter': 1}), 'Kim Director': defaultdic
t(<class 'int'>, {'Joe Berlinger': 1}), 'Christine Taylor': defaultdict(<class 'i
nt'>, {'Andrew Fleming': 1}), 'Sierra McCormick': defaultdict(<class 'int'>, {'El
izabeth Allen Rosenbaum': 1}), 'Kieran Culkin': defaultdict(<class 'int'>, {'Robe
rt Harmon': 1, 'Peter Care': 1, 'Burr Steers': 1, 'Peter Chelsom': 1}), 'Jeff Kob
er': defaultdict(<class 'int'>, {'Martin Weisz': 1}), 'William Hurt': defaultdict
(<class 'int'>, {'Jay Russell': 1, 'L  a Pool': 1, 'Stephen Hopkins': 1, 'Randall
Miller': 1}), 'Kelly Carlson': defaultdict(<class 'int'>, {'John Bonito': 1}), 'J
ason Priestley': defaultdict(<class 'int'>, {'Stephan Elliott': 1, 'Richard Kwiet
niowski': 1}), 'Yuri Lowenthal': defaultdict(<class 'int'>, {'Mike Nawrocki':

```

```

1)), 'Daniel Stern': defaultdict(<class 'int'>, {'Drew Barrymore': 1, 'Douglas Cheek': 1}), 'Trevor Morgan': defaultdict(<class 'int'>, {'Steve Gomer': 1}), 'Oliver James': defaultdict(<class 'int'>, {'Sean McNamara': 1}), 'David Ogden Stiers': defaultdict(<class 'int'>, {'': 2}), 'Khandi Alexander': defaultdict(<class 'int'>, {'Ron Shelton': 1, 'Leon Ichaso': 1}), 'Elizabeth Berrington': defaultdict(<class 'int'>, {'Martin McDonagh': 1}), 'Larry David': defaultdict(<class 'int'>, {'Woody Allen': 1, 'Joshua Tickell': 1}), 'Matthew Settle': defaultdict(<class 'int'>, {'Mary Lambert': 1}), 'Levon Helm': defaultdict(<class 'int'>, {'Tommy Lee Jones': 1}), 'Anna Silk': defaultdict(<class 'int'>, {'': 1}), 'Thomas Kretschmann': defaultdict(<class 'int'>, {'Brad Anderson': 1, 'Oliver Hirschbiegel': 1}), 'Curtis Armstrong': defaultdict(<class 'int'>, {'Michael Chapman': 1, 'Doug Atchison': 1, 'Stephen Sommers': 1}), 'Thomas Stoppel': defaultdict(<class 'int'>, {'Khalil Sullins': 1}), 'Brigitte Helm': defaultdict(<class 'int'>, {'Fritz Lang': 1}), 'David Belle': defaultdict(<class 'int'>, {'Pierre Morel': 1}), 'Qi Shu': defaultdict(<class 'int'>, {'Hsiao-Hsien Hou': 1}), 'Konstantin Lavronenko': defaultdict(<class 'int'>, {'Andrey Zvyagintsev': 1}), 'Nirut Sirichanya': defaultdict(<class 'int'>, {'Tony Jaa': 1}), 'Christopher Heyerdahl': defaultdict(<class 'int'>, {'Russell Mulcahy': 1}), 'Eddie Izzard': defaultdict(<class 'int'>, {'Stefan Ruzowitzky': 1, 'E. Elias Merhige': 1}), 'Stephen Graham': defaultdict(<class 'int'>, {'Jake Paltrow': 1, 'Roland Suso Richter': 1, 'Tom Hooper': 1, 'Shane Meadows': 1}), 'Anna Friel': defaultdict(<class 'int'>, {'Juraj Jakubisko': 1, 'David Leland': 1, 'Barry Levinson': 1}), 'Birgitte Hjort Sørensen': defaultdict(<class 'int'>, {'Gabe Ibáñez': 1}), 'Bruce Payne': defaultdict(<class 'int'>, {'Gerry Lively': 1}), 'Maximilian Dirr': defaultdict(<class 'int'>, {'Daniele Lucchetti': 1}), 'Samantha Morton': defaultdict(<class 'int'>, {'Julien Temple': 1, 'Michael Winterbottom': 1, 'Lynne Ramsay': 1, 'Liv Ullmann': 1, 'Eric Styles': 1}), 'Sharon Leal': defaultdict(<class 'int'>, {'Peter Cousens': 1}), 'Anil Kapoor': defaultdict(<class 'int'>, {'Danny Boyle': 1}), 'Fred Gwynne': defaultdict(<class 'int'>, {'Adrian Lyne': 1, 'Jonathan Lynn': 1}), 'Marxada Valverde': defaultdict(<class 'int'>, {'Vidhu Vinod Chopra': 1}), 'Doug E. Doug': defaultdict(<class 'int'>, {'Jon Turteltaub': 1}), 'Jensen Ackles': defaultdict(<class 'int'>, {'Patrick Lussier': 1, '': 1}), 'Caroline Munro': defaultdict(<class 'int'>, {'Lewis Gilbert': 1}), 'Amy Bruni': defaultdict(<class 'int'>, {'': 1}), 'Phil Vischer': defaultdict(<class 'int'>, {'Mike Nawrocki': 1}), 'Janet Jackson': defaultdict(<class 'int'>, {'John Singleton': 1}), 'Mike Epps': defaultdict(<class 'int'>, {'Kevin Bray': 1, 'Marcus Raboy': 1}), 'Linda Blair': defaultdict(<class 'int'>, {'John Boorman': 1}), 'Lily Tomlin': defaultdict(<class 'int'>, {'Franco Zeffirelli': 1, 'David O. Russell': 1}), 'Peter Mensah': defaultdict(<class 'int'>, {'James Isaac': 1, '': 1}), 'Monica Potter': defaultdict(<class 'int'>, {'Mark Waters': 1}), 'Thomas McDonell': defaultdict(<class 'int'>, {'Josh Schwartz': 1}), 'Emmanuelle Seigner': defaultdict(<class 'int'>, {'Julian Schnabel': 1}), 'Derek Luke': defaultdict(<class 'int'>, {'Phillip Noyce': 1}), 'Melissa Sagemiller': defaultdict(<class 'int'>, {'Stephen Carpenter': 1}), 'Seth Gilliam': defaultdict(<class 'int'>, {'James Ivory': 1}), 'Lesley Manville': defaultdict(<class 'int'>, {'Mike Leigh': 2}), 'Mathieu Amalric': defaultdict(<class 'int'>, {'Alain Resnais': 1, 'Mathieu Amalric': 1}), 'Sebastian Koch': defaultdict(<class 'int'>, {'Costa-Gavras': 1, 'Florian Henckel von Donnersmarck': 1}), 'Woo-sung Jung': defaultdict(<class 'int'>, {'Chao-Bin Su': 1}), 'John Heard': defaultdict(<class 'int'>, {'Neil Burger': 1, 'Ed Harris': 1}), 'Krystyna Janda': defaultdict(<class 'int'>, {'': 1}), 'Jacek Koman': defaultdict(<class 'int'>, {'': 1}), 'Belón Rueda': defaultdict(<class 'int'>, {'Alejandro Amenábar': 1}), 'Jason Mewes': defaultdict(<class 'int'>, {'Hyung-rae Shim': 1, 'Kevin Smith': 1}), 'Usher Raymond': defaultdict(<class 'int'>, {'Jon M. Chu': 1}), 'Sean Patrick Thomas': defaultdict(<class 'int'>, {'Thomas Carter': 1, 'Tim Story': 1}), 'Tuesday Knight': defaultdict(<class 'int'>, {'Renny Harlin': 1}), 'Jack Palance': defaultdict(<class 'int'>, {'Christopher Cain': 1}), 'Scar Jaenada': defaultdict(<class 'int'>, {'Jaume Collat-Serra': 1}), 'Chris Elliott': defaultdict(<class 'int'>, {'Chris Koch': 1}), 'Kathleen Turner': defaultdict(<class 'int'>, {'Bob Clark': 1, 'John Waters': 1}), 'Jenny Agutter': defaultdict(<class 'int'>, {'John Lafia': 1}), 'Elina Alminas': defaultdict(<class 'int'>, {'Alex Garland': 1}), 'Teo Halm': defaultdict(<cl

```

```

ass 'int'>, {'Dave Green': 1}}, 'Yuki Matsuzaki': defaultdict(<class 'int'>, {'Cl
int Eastwood': 1}}, 'Don Rickles': defaultdict(<class 'int'>, {'Bob Saget': 1}},
'Michael Nyqvist': defaultdict(<class 'int'>, {'Niels Arden Oplev': 1, 'Kay Polla
k': 1}}, 'Justin Chon': defaultdict(<class 'int'>, {'Jon Lucas': 1, 'Alex Craig M
ann': 1}}, 'Burt Young': defaultdict(<class 'int'>, {'Jon Hess': 1}}, 'John Boyeg
a': defaultdict(<class 'int'>, {'Joe Cornish': 1}}, 'Jelena Jovanova': defaultdic
t(<class 'int'>, {'Angelina Jolie Pitt': 1}}, 'Steve Irwin': defaultdict(<class
'int'>, {'John Stainton': 1}}, 'Dameon Clarke': defaultdict(<class 'int'>, {'Glen
n Ficarra': 1}}, 'Rae Dawn Chong': defaultdict(<class 'int'>, {'Jean-Jacques Anna
ud': 1}}, 'Gabriel Millman': defaultdict(<class 'int'>, {'Michael Hoffman': 1}},
'Julian Morris': defaultdict(<class 'int'>, {'Stewart Hendler': 1, 'Darin Scott':
1, 'Jeff Wadlow': 1, 'Oliver Blackburn': 1}}, 'David Niven': defaultdict(<class
'int'>, {'John Cromwell': 1}}, 'Richard Johnson': defaultdict(<class 'int'>, {'Ma
rk Herman': 1}}, 'Isabella Rossellini': defaultdict(<class 'int'>, {'Abel Ferrar
a': 1, 'James Gray': 1, 'Paul Fierlinger': 1}}, 'Min-sik Choi': defaultdict(<clas
s 'int'>, {'Je-kyu Kang': 1, 'Chan-wook Park': 2}}, 'Sean Young': defaultdict(<cl
ass 'int'>, {'Tom Shadyac': 1}}, 'Alex Veadov': defaultdict(<class 'int'>, {'Mike
McCoy': 1}}, 'Shelley Duvall': defaultdict(<class 'int'>, {'Terry Gilliam': 1}},
'Britney Spears': defaultdict(<class 'int'>, {'Tamra Davis': 1}}, 'Dax Flame': de
faultdict(<class 'int'>, {'Nima Nourizadeh': 1}}, 'George C. Scott': defaultdict
(<class 'int'>, {'Franklin J. Schaffner': 1, 'Robert Rossen': 1, 'Stanley Kubric
k': 1}}, 'Steven Mackintosh': defaultdict(<class 'int'>, {'Brian Henson': 1}}, 'L
exie Contursi': defaultdict(<class 'int'>, {'Dan Cutforth': 1}}, 'Talia Shire': d
efaultdict(<class 'int'>, {'John Frankenheimer': 1}}, 'Tammin Sursok': defaultdic
t(<class 'int'>, {'Elizabeth Allen Rosenbaum': 1}}, 'Nat Wolff': defaultdict(<cla
ss 'int'>, {'Jake Schreier': 1}}, 'Common': defaultdict(<class 'int'>, {'Sanaa Ha
mri': 1, 'Michel Gondry': 1}}, 'JÃ¼rgen Prochnow': defaultdict(<class 'int'>, {'W
olfgang Petersen': 1}}, 'Heather Matarazzo': defaultdict(<class 'int'>, {'Wallace
Wolodarsky': 1, 'Todd Solondz': 1}}, 'Tom Hughes': defaultdict(<class 'int'>, {'R
ichard Curtis': 1}}, 'Nate Parker': defaultdict(<class 'int'>, {'Nicholas Jareck
i': 1, 'Gina Prince-Bythewood': 1}}, 'Gary Weeks': defaultdict(<class 'int'>, {'D
ean Israelite': 1}}, 'Melanie Griffith': defaultdict(<class 'int'>, {'Woody Alle
n': 1, 'Brian De Palma': 1}}, 'Jill Clayburgh': defaultdict(<class 'int'>, {'Ryan
Murphy': 1}}, 'Anika Noni Rose': defaultdict(<class 'int'>, {'Robert Iscove':
1}}, 'Doona Bae': defaultdict(<class 'int'>, {'Joon-ho Bong': 1}}, 'Jeremy Davie
s': defaultdict(<class 'int'>, {'Antonia Bird': 1, 'Steven Shainberg': 1}}, 'Toni
Servillo': defaultdict(<class 'int'>, {'Paolo Sorrentino': 1}}, 'Mia Wasikowska':
defaultdict(<class 'int'>, {'Chan-wook Park': 1, 'Rodrigo GarcÃaxada': 1}}, 'Davi
d Richmond-Peck': defaultdict(<class 'int'>, {'Ira Sachs': 1}}, 'Eamonn Walker':
defaultdict(<class 'int'>, {'Carroll Ballard': 1}}, 'Eva Amurri Martino': default
dict(<class 'int'>, {'Vadim Perelman': 1}}, 'Marcell Nagy': defaultdict(<class 'i
nt'>, {'Lajos Koltai': 1}}, 'Honglei Sun': defaultdict(<class 'int'>, {'Yimou Zha
ng': 1}}, 'David Hewlett': defaultdict(<class 'int'>, {'Vincenzo Natali': 3}}, 'Y
e Liu': defaultdict(<class 'int'>, {'Chuan Lu': 1}}, 'Richard Coyle': defaultdict
(<class 'int'>, {'Renny Harlin': 1}}, 'Rachael Carpani': defaultdict(<class 'in
t'>, {'Christopher Smith': 1}}, 'Jacob Vargas': defaultdict(<class 'int'>, {'Rand
all Wallace': 1}}, 'Gary Daniels': defaultdict(<class 'int'>, {'David Winters':
1}}, 'Miko Hughes': defaultdict(<class 'int'>, {'Mary Lambert': 1, 'Wes Craven':
1}}, 'Tatsuo Matsumura': defaultdict(<class 'int'>, {'Akira Kurosawa': 1}}, 'Wei
Tang': defaultdict(<class 'int'>, {'Mabel Cheung': 1}}, 'Phoebe Cates': defaultdi
ct(<class 'int'>, {'Joe Dante': 1}}, 'Wagner Moura': defaultdict(<class 'int'>,
{'Stephen Daldry': 1, 'JosÃ Padilha': 1, 'Karim AÃnouz': 1}}, 'Lucas Grabeel':
defaultdict(<class 'int'>, {'Kenny Ortega': 3}}, 'Zoe Ventoura': defaultdict(<cla
ss 'int'>, {'Brian Trenchard-Smith': 1}}, 'Corbin Bernsen': defaultdict(<class 'i
nt'>, {'David S. Ward': 1, 'Rusty Cundieff': 1, '': 1, 'Corbin Bernsen': 1, 'Dani
el Mellitz': 1}}, 'Lily Cole': defaultdict(<class 'int'>, {'Oliver Parker': 1}},
'Lauren German': defaultdict(<class 'int'>, {'Adam Shankman': 1}}, 'Lois Smith':
defaultdict(<class 'int'>, {'Tim Robbins': 1}}, 'Madeline Kahn': defaultdict(<cla
ss 'int'>, {'Mel Brooks': 4, 'James Frawley': 1}}, 'Steven Bauer': defaultdict(<c
lass 'int'>, {'Brian De Palma': 1}}, 'Louise Fletcher': defaultdict(<class 'in

```

```

t'}, {'Tobe Hooper': 1}), 'Lee Majors': defaultdict(<class 'int'>, {'Brendan Mall
oy': 1}), 'Luke Newberry': defaultdict(<class 'int'>, {'Dustin Hoffman': 1}), 'Pa
tricia Tallman': defaultdict(<class 'int'>, {'Sam Raimi': 1}), 'Michael McKean':
defaultdict(<class 'int'>, {'Daisy von Scherler Mayer': 1, 'Christopher Guest':
1}), 'Phyllida Law': defaultdict(<class 'int'>, {'Agnieszka Holland': 1}), 'Sam R
iley': defaultdict(<class 'int'>, {'Rowan Joffe': 1, 'Anton Corbijn': 1}), 'John
Cassavetes': defaultdict(<class 'int'>, {'Giuliano Montaldo': 1, 'Roman Polansk
i': 1}), 'Nora Dunn': defaultdict(<class 'int'>, {'Lisa Azuelos': 1}), 'John Howa
rd': defaultdict(<class 'int'>, {'Ray Lawrence': 1}), 'Shah Rukh Khan': defaultdi
ct(<class 'int'>, {'Karan Johar': 2, 'Yash Chopra': 2}), 'Keir Dullea': defaultdi
ct(<class 'int'>, {'Stanley Kubrick': 1}), 'Henry Thomas': defaultdict(<class 'in
t'>, {'Steven Spielberg': 1, 'Greg Marcks': 1, 'Michael Landon Jr.': 1}), 'Ricky
Schroder': defaultdict(<class 'int'>, {'Floyd Mutrux': 1, 'Jeb Stuart': 1}), 'Ant
onia Thomas': defaultdict(<class 'int'>, {'': 1}), 'Arjun Rampal': defaultdict(<c
lass 'int'>, {'Sajid Khan': 1}), 'Jon Gries': defaultdict(<class 'int'>, {'Christ
opher Cain': 1}), 'Santiago Cabrera': defaultdict(<class 'int'>, {'Dean Wright':
1}), 'Karin Viard': defaultdict(<class 'int'>, {'Eric Lartigau': 1}), 'Boris Kodj
oe': defaultdict(<class 'int'>, {'Tyler Perry': 1, 'Douglas Aarniokoski': 1, 'Ric
k Famuyiwa': 1, 'Bille Woodruff': 1}), 'Richard Burton': defaultdict(<class 'in
t'>, {'Ken Annakin': 1, 'Henry Koster': 1}), 'Lane Smith': defaultdict(<class 'in
t'>, {'Stephen Herek': 1}), 'Tina Turner': defaultdict(<class 'int'>, {'George Mi
ller': 1}), 'Lauren Cohan': defaultdict(<class 'int'>, {'William Brent Bell':
1}), 'Jeff Fahey': defaultdict(<class 'int'>, {'Brett Leonard': 1}), 'Laila Hale
y': defaultdict(<class 'int'>, {'Ciar n Foy': 1}), 'Billy Drago': defaultdict(<c
lass 'int'>, {'Joseph Zito': 1}), 'Jurnee Smollett-Bell': defaultdict(<class 'in
t'>, {'Malcolm D. Lee': 1, 'Kasi Lemmons': 1}), 'David Zayas': defaultdict(<class
'int'>, {'Colin Strause': 1}), 'Tina Desai': defaultdict(<class 'int'>, {'John Ma
dden': 1}), 'Ben Feldman': defaultdict(<class 'int'>, {'Mark Rosman': 1, 'John Er
ick Dowdle': 1}), 'Steve Oedekerk': defaultdict(<class 'int'>, {'Steve Oedekerk':
1}), 'Reba McEntire': defaultdict(<class 'int'>, {'Ron Underwood': 1}), 'Byron Ma
nn': defaultdict(<class 'int'>, {'James Foley': 1}), 'Wendi McLendon-Covey': defa
ultdict(<class 'int'>, {'Robert Ben Garant': 1}), 'Harry Styles': defaultdict(<cl
ass 'int'>, {'Morgan Spurlock': 1}), 'Chase Ellison': defaultdict(<class 'int'>,
{'Jim Hanon': 1}), 'Jasper Cole': defaultdict(<class 'int'>, {'Jorma Taccone':
1}), 'Sophie Okonedo': defaultdict(<class 'int'>, {'Stephen Frears': 1}), 'Michae
la Conlin': defaultdict(<class 'int'>, {'': 1}), 'Daniel Sharman': defaultdict(<c
lass 'int'>, {'Marcus Dunstan': 1}), 'Johannes Silberschneider': defaultdict(<cla
ss 'int'>, {'Fran ois Girard': 1}), 'Dylan Moran': defaultdict(<class 'int'>,
{'David Schwimmer': 1}), 'Bernadette Peters': defaultdict(<class 'int'>, {'Allan
Arkush': 1}), 'Flavio Parenti': defaultdict(<class 'int'>, {'Luca Guadagnino':
1}), 'Robert Picardo': defaultdict(<class 'int'>, {'John Putsch': 1}), 'Peter Fond
a': defaultdict(<class 'int'>, {'Steven Soderbergh': 1, 'Victor Nunez': 1}), 'Eri
c Stoltz': defaultdict(<class 'int'>, {'Terence Davies': 1, 'Roger Avary': 1}),
'Cliff Robertson': defaultdict(<class 'int'>, {'Harley Cokeliss': 1, 'Ralph Nelso
n': 1}), 'Scott Mechlowicz': defaultdict(<class 'int'>, {'Victor Salva': 1}), 'Gi
llian White': defaultdict(<class 'int'>, {'Spike Lee': 1}), 'Eoin Macken': defaul
tdict(<class 'int'>, {'Jason Zada': 1}), 'Miranda Richardson': defaultdict(<class
'int'>, {'David Cronenberg': 1}), 'Naseeruddin Shah': defaultdict(<class 'int'>,
{'Rakesh Roshan': 1, 'Mira Nair': 1}), 'Bashar Rahal': defaultdict(<class 'int'>,
{'Joshua Seftel': 1, 'Uwe Boll': 1}), 'Wei Zhao': defaultdict(<class 'int'>, {'St
ephens Chow': 1}), 'Vincent Gallo': defaultdict(<class 'int'>, {'Vincent Gallo':
1}), 'Isabelle Huppert': defaultdict(<class 'int'>, {'Claude Chabrol': 1, 'Michae
l Haneke': 1}), 'Numan Acar': defaultdict(<class 'int'>, {'Jon Stewart': 1}), 'Ki
p Pardue': defaultdict(<class 'int'>, {'Dan Harris': 1}), 'Danny Dyer': defaultdi
ct(<class 'int'>, {'Christopher Smith': 1, 'Justin Kerrigan': 1, 'Dan Reed': 1}),
'Jorge Luis Moreno': defaultdict(<class 'int'>, {'Rafa Lara': 1}), 'Adrian Martin
ez': defaultdict(<class 'int'>, {'Julio DePietro': 1}), 'Stephanie Szostak': defa
ultdict(<class 'int'>, {'Katherine Dieckmann': 1, 'Paolo Monico': 1}), 'Madison P
ettis': defaultdict(<class 'int'>, {'William Dear': 1}), 'Nicholas Hamilton': def
aultdict(<class 'int'>, {'Kim Farrant': 1}), 'Drew Fuller': defaultdict(<class 'i

```

```

nt'}, {'Scott Marshall': 1}}, 'Jim Carter': defaultdict(<class 'int'>, {'Ã\x81lex
de la Iglesia': 1}}, 'Damian Walshe-Howling': defaultdict(<class 'int'>, {'Andrew
Traucki': 1}}, 'Teryl Rothery': defaultdict(<class 'int'>, {'Patrick Lussier':
1}}, 'Shane Pollard': defaultdict(<class 'int'>, {'Robert Adetuyi': 1}}, 'Veronic
a Ferres': defaultdict(<class 'int'>, {'Paul Schrader': 1}}, 'Ingvar Eggert Sigur
Ãsson': defaultdict(<class 'int'>, {'Benedikt Erlingsson': 1}}, 'Sendhil Ramamur
thy': defaultdict(<class 'int'>, {'Gurinder Chadha': 1, '': 1}}, 'Jimi Mistry': d
efaultdict(<class 'int'>, {'Vic Sarin': 1}}, 'Kang-ho Song': defaultdict(<class
'int'>, {'Jee-woon Kim': 1}}, 'Tyler Mane': defaultdict(<class 'int'>, {'William
Phillips': 1}}, 'Danny Pino': defaultdict(<class 'int'>, {'Andy Garcia': 1, '':
1}}, 'Daniella Alonso': defaultdict(<class 'int'>, {'Valeri Milev': 1}}, 'Carmen
Maura': defaultdict(<class 'int'>, {'Pedro AlmodÃ³var': 1}}, 'John Vernon': defau
ltdict(<class 'int'>, {'Gerald Potterton': 1}}, 'Jizelle Jade': defaultdict(<clas
s 'int'>, {'Jared Hess': 1}}, 'BÃ¼rbara Mori': defaultdict(<class 'int'>, {'Anura
g Basu': 1}}, 'Noah Taylor': defaultdict(<class 'int'>, {'Kriv Stenders': 1, 'Sco
tt Hicks': 1}}, 'Topol': defaultdict(<class 'int'>, {'Norman Jewison': 1}}, 'Ricc
ardo Scamarcio': defaultdict(<class 'int'>, {'Ferzan Ozpetek': 1, 'MaÃwenn':
1}}, 'Catherine Hicks': defaultdict(<class 'int'>, {'Tom Holland': 1}}, 'Michael
Moore': defaultdict(<class 'int'>, {'Michael Moore': 4}}, 'Noel Gugliemi': defaul
tdict(<class 'int'>, {'James DeMonaco': 1, 'Youssef Delara': 2, 'Jonas Ã\x85kerlu
nd': 1}}, 'Don Ameche': defaultdict(<class 'int'>, {'John Landis': 1}}, 'Kristin
Lehman': defaultdict(<class 'int'>, {'Christopher McQuarrie': 1}}, 'Ronn Moss': d
efaultdict(<class 'int'>, {'': 1}}, 'Jung-min Hwang': defaultdict(<class 'int'>,
{'Hong-jin Na': 1}}, 'Saruny Wongkrachang': defaultdict(<class 'int'>, {'Chatric
halerm Yukol': 1}}, 'Lambert Wilson': defaultdict(<class 'int'>, {'Eric Lavaine':
1, 'Xavier Beauvois': 1}}, 'Q'orianka Kilcher': defaultdict(<class 'int'>, {'Marc
Forby': 1, 'Hank Braxtan': 1}}, 'Jacqueline McKenzie': defaultdict(<class 'int'>,
{'Peter Cattaneo': 1}}, 'Lars Mikkelsen': defaultdict(<class 'int'>, {'Ole Christ
ian Madsen': 1}}, 'Harrison Gilbertson': defaultdict(<class 'int'>, {'Jeremy Sim
s': 1}}, 'Dee Bradley Baker': defaultdict(<class 'int'>, {'': 1}}, 'David Keith':
defaultdict(<class 'int'>, {'Mark Rydell': 1, 'Brian Trenchard-Smith': 1}}, 'Jane
Fonda': defaultdict(<class 'int'>, {'Roger Vadim': 1}}, 'BjÃrn Sundquist': defau
ltdict(<class 'int'>, {'Nils Gaup': 1, 'Tommy Wirkola': 1}}, 'Jacinda Barrett': d
efaultdict(<class 'int'>, {'Mira Nair': 1}}, 'Mariah Carey': defaultdict(<class
'int'>, {'Vondie Curtis-Hall': 1}}, 'Abigail Spencer': defaultdict(<class 'int'>,
{'Tom Elkins': 1}}, 'Yu-mi Jeong': defaultdict(<class 'int'>, {'Woo-Suk Kang':
1}}, 'Robert Redford': defaultdict(<class 'int'>, {'J.C. Chandor': 1}}, 'Mary Eli
zabeth Mastrantonio': defaultdict(<class 'int'>, {'John Sayles': 1}}, 'Katrina Ka
if': defaultdict(<class 'int'>, {'Vipul Amrutlal Shah': 1}}, 'Padraic Delaney': d
efaultdict(<class 'int'>, {'Ken Loach': 1}}, 'Ranbir Kapoor': defaultdict(<class
'int'>, {'Ayan Mukerji': 1, 'Shimit Amin': 1}}, 'Peter Ustinov': defaultdict(<cla
ss 'int'>, {'Mervyn LeRoy': 1}}, 'Necati Sasmaz': defaultdict(<class 'int'>, {'Se
rdar Akar': 1}}, 'Eleanor Parker': defaultdict(<class 'int'>, {'Robert Wise':
1}}, 'Marshall Bell': defaultdict(<class 'int'>, {'Rob Reiner': 1}}, 'Noah Huntle
y': defaultdict(<class 'int'>, {'Danny Boyle': 1}}, 'Jennifer Freeman': defaultdi
ct(<class 'int'>, {'Chris Stokes': 1}}, 'James Kyson': defaultdict(<class 'int'>,
{'Masayuki Ochiai': 1}}, 'Mackenzie Davis': defaultdict(<class 'int'>, {'Tom Gorm
ican': 1}}, 'Brian Doyle-Murray': defaultdict(<class 'int'>, {'Ken Shapiro': 1}},
'Tina Gilton': defaultdict(<class 'int'>, {'Justin Tipping': 1}}, 'Telly Savala
s': defaultdict(<class 'int'>, {'Peter R. Hunt': 1}}, 'Nick Wechsler': defaultdic
t(<class 'int'>, {'': 1}}, 'Stephen Collins': defaultdict(<class 'int'>, {'John S
chultz': 1}}, 'Alexandra Maria Lara': defaultdict(<class 'int'>, {'James Ivory':
1}}, 'Christopher Nicholas Smith': defaultdict(<class 'int'>, {'Nicole Holofcene
r': 1}}, 'Cameron Monaghan': defaultdict(<class 'int'>, {'Joe Nussbaum': 1}}, 'Fa
ye Dunaway': defaultdict(<class 'int'>, {'Michael Winner': 1, 'Frank Perry': 1}},
'Sean Lennon': defaultdict(<class 'int'>, {'Tomm Moore': 1}}, 'Alexia Fast': defa
ultdict(<class 'int'>, {'Andrew Currie': 1}}, 'Ian Tracey': defaultdict(<class 'i
nt'>, {'Claudia Llosa': 1}}, 'Mitsuo Iwata': defaultdict(<class 'int'>, {'Katsuhi
ro Ã\x94tomo': 1}}, 'Stephen McHattie': defaultdict(<class 'int'>, {'Charles Bina
mÃ': 1, 'Bruce McDonald': 1, 'Boris Rodriguez': 1}}, 'Rachel Ann Mullins': defau

```



```

ltdict(<class 'int'>, {'Keith Parmer': 1}), 'Leehom Wang': defaultdict(<class 'int'>, {'Dennie Gordon': 1}), 'Celia Imrie': defaultdict(<class 'int'>, {'Ol Parker': 1}), 'Mckenna Grace': defaultdict(<class 'int'>, {'Bruce Beresford': 1}), 'Spencer Wilding': defaultdict(<class 'int'>, {'James Nunn': 1}), 'Scott Takeda': defaultdict(<class 'int'>, {'Michael Winnick': 1}), 'Fran soise Yip': defaultdict(<class 'int'>, {'Stanley Tong': 1}), 'Joely Richardson': defaultdict(<class 'int'>, {'Tony Richardson': 1, 'Henry Hobson': 1}), 'Molly Parker': defaultdict(<class 'int'>, {'Paul Gross': 1}), 'Eric Johnson': defaultdict(<class 'int'>, {'Russell Holt': 1}), 'Sandrine Kiberlain': defaultdict(<class 'int'>, {'Claude Miller': 1}), 'Akon': defaultdict(<class 'int'>, {'Jeta Amata': 1}), 'Saif Ali Khan': defaultdict(<class 'int'>, {'Siddharth Anand': 1, 'Jugal Hansraj': 1}), 'Max Riemelt': defaultdict(<class 'int'>, {'Dennis Gansel': 1}), 'George Coe': defaultdict(<class 'int'>, {'Robert Marcarelli': 1}), 'Darren Shaharvi': defaultdict(<class 'int'>, {'Ernie Barbarash': 1}), 'Michael Nouri': defaultdict(<class 'int'>, {'Adrian Lyne': 1}), 'Sullivan Stapleton': defaultdict(<class 'int'>, {'Jonathan Liebesman': 1}), 'MyAnna Buring': defaultdict(<class 'int'>, {'Neil Marshall': 1, 'Phil Claydon': 1, 'Ben Wheatley': 1}), 'Sid Haig': defaultdict(<class 'int'>, {'Rob Zombie': 3, 'Ward Roberts': 1}), 'Kimberly Elise': defaultdict(<class 'int'>, {'Rick Famuyiwa': 1}), 'Bernard Fox': defaultdict(<class 'int'>, {'Sharron Miller': 1}), 'Michael Stuhlbarg': defaultdict(<class 'int'>, {'Ethan Coen': 1}), 'Irvin Kershner': defaultdict(<class 'int'>, {'Martin Scorsese': 1}), 'Jonathan Brandis': defaultdict(<class 'int'>, {'Michael Corrente': 1}), 'Indira Varma': defaultdict(<class 'int'>, {'Gurinder Chadha': 1, 'Mira Nair': 1}), 'Roy Billing': defaultdict(<class 'int'>, {'Phillip Noyce': 1}), 'Michael J. Pollard': defaultdict(<class 'int'>, {'Tony Maylam': 1}), 'Melinda Clarke': defaultdict(<class 'int'>, {'': 1, 'Brian Yuzna': 1}), 'Ezra Miller': defaultdict(<class 'int'>, {'Lynne Ramsay': 1, 'Raymond De Felitta': 1}), 'Samantha Barks': defaultdict(<class 'int'>, {'John Stephenson': 1}), 'Nathalie Fay': defaultdict(<class 'int'>, {'Sofia Coppola': 1}), 'Geoff Stults': defaultdict(<class 'int'>, {'Bob Gosse': 1, 'Jennifer Finnigan': 1}), 'Lynn Cohen': defaultdict(<class 'int'>, {'Shari Springer Berman': 1, 'Frank Whaley': 1}), 'Leonid Yarmolnik': defaultdict(<class 'int'>, {'Aleksey German': 1}), 'Michael Wong': defaultdict(<class 'int'>, {'David Lam': 1}), 'Erin Fitzgerald': defaultdict(<class 'int'>, {'Vladlen Barbe': 1}), 'Debi Derryberry': defaultdict(<class 'int'>, {'Richard Rich': 1}), 'Ben Browder': defaultdict(<class 'int'>, {'Robert C. Cooper': 1}), 'Gordon Pinsent': defaultdict(<class 'int'>, {'Kim Nguyen': 1}), 'Rex Baker': defaultdict(<class 'int'>, {'Jackie Earle Haley': 1}), 'Heike Makatsch': defaultdict(<class 'int'>, {'Max F rberb ck': 1}), 'Glenn Plummer': defaultdict(<class 'int'>, {'Damian Nieman': 1}), 'Lubna Azabal': defaultdict(<class 'int'>, {'Denis Villeneuve': 1}), 'Laura Morante': defaultdict(<class 'int'>, {'Gabriele Muccino': 1}), 'Deborah Mailman': defaultdict(<class 'int'>, {'Rachel Perkins': 1}), 'John Cothran': defaultdict(<class 'int'>, {'John Singleton': 1}), 'Rita Moreno': defaultdict(<class 'int'>, {'Alan Alda': 1, 'Jerome Robbins': 1, 'Stanley Donen': 1}), 'Angie Dickinson': defaultdict(<class 'int'>, {'Brian De Palma': 1}), 'Greg Germann': defaultdict(<class 'int'>, {'Nicole Holofcener': 1}), 'David Wayne': defaultdict(<class 'int'>, {'Robert Wise': 1}), 'Leon': defaultdict(<class 'int'>, {'Louis Morneau': 1}), 'Merab Ninidze': defaultdict(<class 'int'>, {'Caroline Link': 1}), 'Emily Podleski': defaultdict(<class 'int'>, {'Sterling Van Wagenen': 1}), 'Florian Lukas': defaultdict(<class 'int'>, {'Wolfgang Becker': 1}), 'Linnea Quigley': defaultdict(<class 'int'>, {'Dan O'Bannon': 1, 'Herb Freed': 1}), 'Maria Pia Calzone': defaultdict(<class 'int'>, {'': 1}), 'Olivia Newton-John': defaultdict(<class 'int'>, {'Randal Kleiser': 1}), 'Osama bin Laden': defaultdict(<class 'int'>, {'Michael Moore': 1}), 'Katharine Ross': defaultdict(<class 'int'>, {'George Roy Hill': 1, 'Michael D. Sellers': 1}), 'Ed Wynn': defaultdict(<class 'int'>, {'Robert Stevenson': 1}), 'Lisa Wilcox': defaultdict(<class 'int'>, {'Stephen Hopkins': 1}), 'Tisha Campbell-Martin': defaultdict(<class 'int'>, {'Spike Lee': 1}), 'Sammo Kam-Bo Hung': defaultdict(<class 'int'>, {'Sammo Kam-Bo Hung': 1}), 'Hugh M. Hefner': defaultdict(<class 'int'>, {'Zach Cregger': 1, 'Fenton Bailey': 1}), 'Francis Capra': defaultdict(<class 'int'>, {'': 1}), 'John Cameron Mitchell': defaultdict(<class 'int'>, {'John Cameron Mitchell': 1}), 'Antoni Corone': defaultdict(<class 'int'>, {'Sheldon Lettich': 1}),

```

```

'Rachel Alig': defaultdict(<class 'int'>, {'Jon Knautz': 1}), 'Chelcie Ross': def
aultdict(<class 'int'>, {'David Atkins': 1}), 'William Holden': defaultdict(<clas
s 'int'>, {'Sam Peckinpah': 1, 'David Lean': 1}), 'Lesley-Anne Down': defaultdict
(<class 'int'>, {'Michael Crichton': 1}), 'Michael Berryman': defaultdict(<class
'int'>, {'Sylvio Tabet': 1, 'Ruggero Deodato': 1, 'Justin Thomas Ostensen': 1}),
'Benoît Magimel': defaultdict(<class 'int'>, {'Claude Chabrol': 1}), 'James Nesb
itt': defaultdict(<class 'int'>, {'Peter Cattaneo': 1, '': 1, 'Kirk Jones': 1, 'P
aul Greengrass': 1}), 'Noam Chomsky': defaultdict(<class 'int'>, {'Tony Kaye': 1,
'Sut Jhally': 1}), 'Catherine Bell': defaultdict(<class 'int'>, {'Perry Lang':
1}), 'Pruitt Taylor Vince': defaultdict(<class 'int'>, {'Jake Goldberger': 1}),
'Drew Waters': defaultdict(<class 'int'>, {'William Kaufman': 1}), 'Debbie Harr
y': defaultdict(<class 'int'>, {'David Cronenberg': 1}), 'Yorick van Wageningen':
defaultdict(<class 'int'>, {'Martin Koolhoven': 1}), 'Jeanne Moreau': defaultdict
(<class 'int'>, {'John Frankenheimer': 1}), 'Simon Bird': defaultdict(<class 'in
t'>, {'': 1}), 'Wesley Jonathan': defaultdict(<class 'int'>, {'Preston A. Whitmor
e II': 1, 'Michael Taliferro': 1}), 'Asin': defaultdict(<class 'int'>, {'Ashish
R. Mohan': 1}), 'Jonathan D. Mellor': defaultdict(<class 'int'>, {'Jaume Balaguer
Ã³': 1}), 'Kayla Jackson': defaultdict(<class 'int'>, {'Stewart Raffill': 1}), 'A
lice Krige': defaultdict(<class 'int'>, {'Hugh Hudson': 1, 'Amanda Gusack': 1}),
'Stephen Macht': defaultdict(<class 'int'>, {'William Sachs': 1}), 'Jean-Baptiste
Maunier': defaultdict(<class 'int'>, {'Christophe Barratier': 1}), 'Alexis Dzien
a': defaultdict(<class 'int'>, {'James Cox': 1}), 'Ruby Dee': defaultdict(<class
'int'>, {'Spike Lee': 1}), 'Jean-Pierre L  aud': defaultdict(<class 'int'>, {'Aki
Kaurism  ki': 1}), 'Philippe Nahon': defaultdict(<class 'int'>, {'Gaspar No  ':
1}), 'Lindsay Kay Hayward': defaultdict(<class 'int'>, {'Hitoshi Matsumoto': 1}),
'Anupam Kher': defaultdict(<class 'int'>, {'Rakeysh Omprakash Mehra': 1}), 'Kim S
haw': defaultdict(<class 'int'>, {'Collin Schiffli': 1}), 'Aksel Hennie': default
dict(<class 'int'>, {'Morten Tyldum': 1}), 'Russell Tovey': defaultdict(<class 'i
nt'>, {'Jon Wright': 1}), 'Campbell Scott': defaultdict(<class 'int'>, {'Michael
McGowan': 1, 'Charles Ferguson': 1}), 'Teresa Wright': defaultdict(<class 'int'>,
{'Jeannot Szwarc': 1}), 'Abhishek Bachchan': defaultdict(<class 'int'>, {'Rohan S
ippy': 1, 'Arjun Sablok': 1}), 'Johanna Braddy': defaultdict(<class 'int'>, {'Hen
ry Joost': 1}), 'John Saxon': defaultdict(<class 'int'>, {'Chuck Russell': 1}),
'Eugenio Derbez': defaultdict(<class 'int'>, {'Eugenio Derbez': 1}), 'Matt Shivel
y': defaultdict(<class 'int'>, {'Henry Joost': 1}), 'Burt Kwouk': defaultdict(<cl
ass 'int'>, {'Blake Edwards': 1}), 'Richard Cabral': defaultdict(<class 'int'>,
{'Christopher Landon': 1}), 'James Lafferty': defaultdict(<class 'int'>, {'Mike F
lanagan': 1}), 'William Schallert': defaultdict(<class 'int'>, {'George Jackson':
1}), 'Tony Bonner': defaultdict(<class 'int'>, {'George Miller': 1}), 'John Beasl
ey': defaultdict(<class 'int'>, {'Robert Duvall': 1}), 'Gene Tierney': defaultdic
t(<class 'int'>, {'Michael Curtiz': 1}), 'Kane Hodder': defaultdict(<class 'in
t'>, {'Rob Hedden': 1, 'Adam Marcus': 1, 'John Carl Buechler': 1, 'Matt Jackson':
1}), 'Maribel Verd  ': defaultdict(<class 'int'>, {'Alfonso Cuar  n': 1}), 'Peter
Serafinowicz': defaultdict(<class 'int'>, {'Edgar Wright': 1}), 'Jessica Lownde
s': defaultdict(<class 'int'>, {'Mickey Liddell': 1, 'Shyam Madiraju': 1}), 'Warr
en Christie': defaultdict(<class 'int'>, {'Gonzalo L  pez-Gallego': 1}), 'Sterlin
g Jerins': defaultdict(<class 'int'>, {'John Erick Dowdle': 1}), 'Vanna Bonta': d
efaultdict(<class 'int'>, {'Don Coscarelli': 1}), 'Gina Lollobrigida': defaultdic
t(<class 'int'>, {'King Vidor': 1, 'Carol Reed': 1}), 'Fred Willard': defaultdict
(<class 'int'>, {'Michael Tiddes': 1}), 'Gary Cooper': defaultdict(<class 'int'>,
{'Frank Borzage': 1, 'Fred Zinnemann': 1}), 'Lara Jill Miller': defaultdict(<clas
s 'int'>, {'Mamoru Hosoda': 1}), 'Marie-Jos  e Croze': defaultdict(<class 'int'>,
{'Denys Arcand': 1}), 'Robert Shaw': defaultdict(<class 'int'>, {'Richard Leste
r': 1, 'Fred Zinnemann': 1, 'Terence Young': 1}), 'Brendan Fehr': defaultdict(<cl
ass 'int'>, {'J.S. Cardone': 1}), 'Fran Drescher': defaultdict(<class 'int'>, {'J
ay Levey': 1}), 'Claudia Cardinale': defaultdict(<class 'int'>, {'Sergio Leone':
1}), 'Norbert Ferrer': defaultdict(<class 'int'>, {'Ronan Chapalain': 1}), 'Cassa
ndra Peterson': defaultdict(<class 'int'>, {'Jerry Belson': 1}), 'Kim Dickens': d
efaultdict(<class 'int'>, {'Jake Kasdan': 1}), 'Dana Delany': defaultdict(<class
'int'>, {'Paul Schrader': 1}), 'Alia Shawkat': defaultdict(<class 'int'>, {'Jerem

```

```

y Saulnier': 1}}, 'Liza Minnelli': defaultdict(<class 'int'>, {'Billy Kent': 1}},
'Andrew Wilson': defaultdict(<class 'int'>, {'Wes Anderson': 1}}, 'Akshaye Khann
a': defaultdict(<class 'int'>, {'Feroz Abbas Khan': 1}}, 'Jeffrey Frost': default
dict(<class 'int'>, {'Errol Morris': 1}}, 'Raymond Burr': defaultdict(<class 'in
t'>, {'Dennis Hopper': 1}}, 'Judith Anderson': defaultdict(<class 'int'>, {'Richa
rd Wallace': 1}}, 'Meghan Ory': defaultdict(<class 'int'>, {'Matthew Hastings':
1}}, 'Ocean James': defaultdict(<class 'int'>, {'M. Night Shyamalan': 1}}, 'Mike
Figueroa': defaultdict(<class 'int'>, {'Brian De Palma': 1}}, 'Glenn Campbell': d
efaultdict(<class 'int'>, {'Oren Peli': 1}}, 'Luis Tosar': defaultdict(<class 'in
t'>, {'Jaume BalaguerÃ³': 1, 'Fernando LeÃ³n de Aranoa': 1}}, 'Dave Legeno': defa
ultdict(<class 'int'>, {'Paul Andrew Williams': 1}}, 'Teri Polo': defaultdict(<cl
ass 'int'>, {'Eric Bross': 1, 'Charles Adelman': 1}}, 'James Frain': defaultdict
(<class 'int'>, {'Vicky Jewson': 1}}, 'David Huddleston': defaultdict(<class 'in
t'>, {'Peter Hyams': 1}}, 'David Raizor': defaultdict(<class 'int'>, {'Mike Mayha
ll': 1}}, 'Mikal Vega': defaultdict(<class 'int'>, {'Stanton Barrett': 1}}, 'Juli
a Jentsch': defaultdict(<class 'int'>, {'JirÃ¡xad Menzel': 1}}, 'Udo Kier': defau
ltdict(<class 'int'>, {'Fatih Akin': 1}}, 'Katy Mixon': defaultdict(<class 'in
t'>, {'Jeff Nichols': 1, 'Jamie Babbit': 1}}, 'Alex Rocco': defaultdict(<class 'i
nt'>, {'Frank LaLoggia': 1}}, 'Bill Johnson': defaultdict(<class 'int'>, {'Tobe H
ooper': 1}}, 'Maria Conchita Alonso': defaultdict(<class 'int'>, {'Gabriela Tagli
avini': 1}}, 'Bhama': defaultdict(<class 'int'>, {'Shekar': 1}}, 'Keith Carradin
e': defaultdict(<class 'int'>, {'Leonard Farlinger': 1, 'Jonathan Parker': 1}},
'Sara Forestier': defaultdict(<class 'int'>, {'Michel Leclerc': 1}}, 'William Kir
cher': defaultdict(<class 'int'>, {'Robert Sarkies': 1}}, 'Brooke Shields': defau
ltdict(<class 'int'>, {'Randal Kleiser': 1}}, 'Royal Dano': defaultdict(<class 'i
nt'>, {'John Huston': 1}}, 'Dan Fogler': defaultdict(<class 'int'>, {'': 1}}, 'Da
na Andrews': defaultdict(<class 'int'>, {'Walter Lang': 1}}, 'MÃ¡xada Maestro': d
efaultdict(<class 'int'>, {'Carlos Saura': 1, 'Lucrecia Martel': 1}}, 'Alex Kendri
ck': defaultdict(<class 'int'>, {'Andrew Erwin': 1, 'Bill Muir': 1, 'Alex Kendri
ck': 1}}, 'Jaime Camil': defaultdict(<class 'int'>, {'Beto GÃ³mez': 1}}, 'Jan Dec
leir': defaultdict(<class 'int'>, {'Mike van Diem': 1}}, 'Rich Pecci': defaultdic
t(<class 'int'>, {'Todd Solondz': 1}}, 'Nicolas Giraud': defaultdict(<class 'in
t'>, {'RenÃ© FÃ©ret': 1}}, 'Simon Yam': defaultdict(<class 'int'>, {'Johnnie To':
1}}, 'Oksana Akinshina': defaultdict(<class 'int'>, {'Lukas Moodysson': 1}}, 'Jim
my Shergill': defaultdict(<class 'int'>, {'Kabir Sadanand': 1}}, 'Katherine McNam
ara': defaultdict(<class 'int'>, {'Peter Deluise': 1}}, 'Sid Caesar': defaultdict
(<class 'int'>, {'Mel Brooks': 1}}, 'Nimrat Kaur': defaultdict(<class 'int'>, {'R
aja Menon': 1}}, 'Fabian Bolin': defaultdict(<class 'int'>, {'Daniel Espinosa':
1}}, 'Vidya Balan': defaultdict(<class 'int'>, {'R. Balki': 1, 'Rajkumar Hirani':
1}}, 'Yangzom Brauen': defaultdict(<class 'int'>, {'Ivan Engler': 1}}, 'Art Hindl
e': defaultdict(<class 'int'>, {'Bob Clark': 1}}, 'Julian Richings': defaultdict
(<class 'int'>, {'George A. Romero': 1, 'Robert Eggers': 1}}, 'Judith O'Dea': def
aultdict(<class 'int'>, {'George A. Romero': 1}}, 'Gloria Grahame': defaultdict(<
class 'int'>, {'Cecil B. DeMille': 1}}, 'Molly C. Quinn': defaultdict(<class 'in
t'>, {'Duane Journey': 1}}, 'Meg Foster': defaultdict(<class 'int'>, {'John Carpe
nter': 1}}, 'Marlee Matlin': defaultdict(<class 'int'>, {'William Arntz': 1}}, 'E
llar Coltrane': defaultdict(<class 'int'>, {'Richard Linklater': 1}}, 'Angell Con
well': defaultdict(<class 'int'>, {'DJ Pooh': 1}}, 'Montgomery Clift': defaultdic
t(<class 'int'>, {'John Huston': 1, 'Howard Hawks': 1, 'Fred Zinnemann': 1}}, 'Ge
raldine Chaplin': defaultdict(<class 'int'>, {'J.A. Bayona': 1}}, 'Ashley Rickard
s': defaultdict(<class 'int'>, {'Michael Tiddes': 1}}, 'Roscoe Lee Browne': defau
ltdict(<class 'int'>, {'Alfred Hitchcock': 1}}, 'Anna Gunn': defaultdict(<class
'int'>, {'Kevin Smith': 1}}, 'Stephanie Leonidas': defaultdict(<class 'int'>, {'D
ave McKean': 1}}, 'Collin Alfredo St. Dic': defaultdict(<class 'int'>, {'Ron Fric
ke': 1}}, 'Kylie Minogue': defaultdict(<class 'int'>, {'Leos Carax': 1}}, 'Peter
McNamara': defaultdict(<class 'int'>, {'Stefan Schwartz': 1}}, 'Lauren Ambrose':
defaultdict(<class 'int'>, {'Robert Lee King': 1}}, 'Hector Kotsifakis': defaultd
ict(<class 'int'>, {'Jorge RamÃ¡xadrez SuÃ¡rez': 1}}, 'Noah Emmerich': defaultdic
t(<class 'int'>, {'David Schwimmer': 1}}, 'Rachel Griffiths': defaultdict(<class
'int'>, {'Sam Miller': 1, '': 1}}, 'Connor Paolo': defaultdict(<class 'int'>, {'J

```

```

im Mickle': 1}}, 'Doug Brochu': defaultdict(<class 'int'>, {'': 1}}, 'Ed Speleer
s': defaultdict(<class 'int'>, {'Julian Gilbey': 1}}, 'Saad Siddiqui': defaultdic
t(<class 'int'>, {'Ruba Nadda': 1}}, 'Jean Harlow': defaultdict(<class 'int'>,
{'Howard Hughes': 1}}, 'Lily Rabe': defaultdict(<class 'int'>, {'Phil Joanou':
1}}, 'Robin McLeavy': defaultdict(<class 'int'>, {'Sean Byrne': 1}}, 'Kristen Qui
ntrall': defaultdict(<class 'int'>, {'Chris Stokes': 1}}, 'Eric Mabijs': defaulttd
ict(<class 'int'>, {'Mark Griffiths': 1, 'Scott Smith': 1}}, 'Rachel Hendrix': de
faultdict(<class 'int'>, {'Bruce Macdonald': 1, 'Andrew Erwin': 1}}, 'Hattie McDa
niel': defaultdict(<class 'int'>, {'Victor Fleming': 1}}, 'Tom Cullen': defaulttdi
ct(<class 'int'>, {'Richard Raymond': 1, 'Andrew Haigh': 1}}, 'Keenan Wynn': defa
ultdict(<class 'int'>, {'George Sidney': 1}}, 'Kayvan Novak': defaultdict(<class
'int'>, {'Christopher Morris': 1}}, 'Fernanda Montenegro': defaultdict(<class 'in
t'>, {'Andrucha Waddington': 1, 'Walter Salles': 1}}, 'Gladys Cooper': defaultdic
t(<class 'int'>, {'Vincente Minnelli': 1}}, 'Alice Olivia Clarke': defaultdict(<c
lass 'int'>, {'Dagur K  rri': 1}}, 'Brenda Vaccaro': defaultdict(<class 'int'>,
{'John Schlesinger': 1}}, 'Peter Graves': defaultdict(<class 'int'>, {'Jim Abraha
ms': 1}}, 'TaRhonda Jones': defaultdict(<class 'int'>, {'': 1}}, 'Slim Pickens':
defaultdict(<class 'int'>, {'Sam Peckinpah': 1}}, 'John Forsythe': defaultdict(<c
lass 'int'>, {'Richard Brooks': 1}}, 'Colleen Dewhurst': defaultdict(<class 'in
t'>, {'Fred Zinnemann': 1}}, 'Bernard Cribbins': defaultdict(<class 'int'>, {'Alf
red Hitchcock': 1}}, 'Joe Estevez': defaultdict(<class 'int'>, {'Michael Martin':
1}}, 'John Mahoney': defaultdict(<class 'int'>, {'Edward Burns': 1}}, 'Ricardo Da
r  xadr  ': defaultdict(<class 'int'>, {'Juan Jos   Campanella': 1, 'Fabi  n Bielin
sky': 1}}, 'Gideon Emery': defaultdict(<class 'int'>, {'Gideon Raff': 1}}, 'Ted R
aimi': defaultdict(<class 'int'>, {'Sam Raimi': 2, 'Bruce Campbell': 1}}, 'Wanda
Sykes': defaultdict(<class 'int'>, {'Louis C.K.': 1}}, 'Elpidia Carrillo': defaul
tdict(<class 'int'>, {'Salvador Carrasco': 1}}, 'Otto Jespersen': defaultdict(<cl
ass 'int'>, {'Andr     x98vredal': 1}}, 'Diahann Carroll': defaultdict(<class 'in
t'>, {'Warren P. Sonoda': 1}}, 'Jane Goodall': defaultdict(<class 'int'>, {'Charl
es Ferguson': 1}}, 'Chris Brochu': defaultdict(<class 'int'>, {'Daniel Petrie J
r.': 1}}, 'Murray Bartlett': defaultdict(<class 'int'>, {'Eldar Rapaport': 1}},
'Corbin Bleu': defaultdict(<class 'int'>, {'Nathan Frankowski': 1}}, 'Thomas Bo L
arsen': defaultdict(<class 'int'>, {'Thomas Vinterberg': 1}}, 'Armando Riesco': d
efaultdict(<class 'int'>, {'Alejandro Monteverde': 1, 'Zach Braff': 1}}, 'Roddy M
cDowall': defaultdict(<class 'int'>, {'Mark L. Lester': 1, 'Don Taylor': 1, 'J. L
ee Thompson': 2, 'John Ford': 1}}, 'Fran  ois B  gaudeau': defaultdict(<class 'in
t'>, {'Laurent Cantet': 1}}, 'Lee Marvin': defaultdict(<class 'int'>, {'John For
d': 2, 'John Boorman': 1}}, 'Catalina Sandino Moreno': defaultdict(<class 'int'>,
{'Joshua Marston': 1}}, 'Honor Blackman': defaultdict(<class 'int'>, {'Guy Hamilt
on': 1}}, 'Sarah Gadon': defaultdict(<class 'int'>, {'Brandon Cronenberg': 1}},
'Donna Reed': defaultdict(<class 'int'>, {'Frank Capra': 1}}, 'Sprague Grayden':
defaultdict(<class 'int'>, {'Tod Williams': 1}}, 'Clu Gulager': defaultdict(<clas
s 'int'>, {'Jack Sholder': 1}}, 'Rhys Wakefield': defaultdict(<class 'int'>, {'Ja
mes DeMonaco': 1}}, 'Danielle Kotch': defaultdict(<class 'int'>, {'Scott Derricks
on': 1}}, 'Nancy O'Dell': defaultdict(<class 'int'>, {'David Raynr': 1}}, 'Burl I
ves': defaultdict(<class 'int'>, {'Richard Brooks': 1}}, 'Linda Harrison': defaul
tdict(<class 'int'>, {'Ted Post': 1}}, 'Veronica Taylor': defaultdict(<class 'in
t'>, {'Kunihiko Yuyama': 1}}, 'Ed Lauter': defaultdict(<class 'int'>, {'Alfred Hi
tchcock': 1}}, 'Fred MacMurray': defaultdict(<class 'int'>, {'Billy Wilder': 1}},
'Tamara Toumanova': defaultdict(<class 'int'>, {'Alfred Hitchcock': 1}}, 'Shirley
Jones': defaultdict(<class 'int'>, {'Richard Brooks': 1}}, 'Maximilian Schell': d
efaultdict(<class 'int'>, {'Stanley Kramer': 1}}, 'Matthew Senreich': defaultdict
(<class 'int'>, {'': 1}}, 'Seamus Davey-Fitzpatrick': defaultdict(<class 'int'>,
{'Richard Linklater': 1}}, 'John Astin': defaultdict(<class 'int'>, {'Christopher
Leitch': 1}}, 'Angus Scrimm': defaultdict(<class 'int'>, {'Don Coscarelli': 1}},
'John Abraham': defaultdict(<class 'int'>, {'Deepa Mehta': 1}}, 'Callie Thorne':
defaultdict(<class 'int'>, {'Peter M. Cohen': 1}}, 'Peter Ostrum': defaultdict(<c
lass 'int'>, {'Mel Stuart': 1}}, 'Kristy Swanson': defaultdict(<class 'int'>, {'T
om Schulman': 1}}, 'Greta Gerwig': defaultdict(<class 'int'>, {'Noah Baumbach':
1, 'Whit Stillman': 1, 'Jay Duplass': 1}}, 'Eddie Griffin': defaultdict(<class 'i

```

```

nt'}, {'George Gallo': 1, 'Dave Meyers': 1, 'Nick Love': 1}), 'Robyn Lively': defaultdict(<class 'int'>, {'David Nixon': 1}), 'Kevin Pollak': defaultdict(<class 'int'>, {'Enrique Begne': 1, 'Rod Lurie': 1}), 'William Morgan Sheppard': defaultdict(<class 'int'>, {'Michael Landon Jr.': 1}), 'Michael Gross': defaultdict(<class 'int'>, {'Tim Heidecker': 1}), 'Susan Hampshire': defaultdict(<class 'int'>, {'': 1}), 'Alicja Bachleda': defaultdict(<class 'int'>, {'Marco Kreuzpaintner': 1}), 'Johnny Pacar': defaultdict(<class 'int'>, {'Kate Connor': 1, 'Matthew R. Anderson': 1}), 'Matt Cohen': defaultdict(<class 'int'>, {'Deon Taylor': 1}), 'Neil Hopkins': defaultdict(<class 'int'>, {'Philip Zlotorynski': 1}), 'Tom Everett Scott': defaultdict(<class 'int'>, {'Francesca Gregorini': 1, 'W.D. Hogan': 1}), 'John Savage': defaultdict(<class 'int'>, {'Marcio Garcia': 1}), 'French Stewart': defaultdict(<class 'int'>, {'Craig Moss': 1}), 'Everett McGill': defaultdict(<class 'int'>, {'Monte Hellman': 1}), 'Ann-Margret': defaultdict(<class 'int'>, {'Frank Capra': 1}), 'Anthony L. Fernandez': defaultdict(<class 'int'>, {'Isaac Florentine': 1}), 'Dyan Cannon': defaultdict(<class 'int'>, {'Susan Seidelman': 1}), 'Olivia Rose Keegan': defaultdict(<class 'int'>, {'Michael Polish': 1}), 'Greg Crowe': defaultdict(<class 'int'>, {'Heidi Ewing': 1}), 'Cécile De France': defaultdict(<class 'int'>, {'Alexandre Aja': 1}), 'Maeve Dermody': defaultdict(<class 'int'>, {'Leon Ford': 1}), 'Isaac Hayes': defaultdict(<class 'int'>, {'Craig Brewer': 1}), 'Nehemiah Persoff': defaultdict(<class 'int'>, {'Billy Wilder': 1}), 'Margaret Hamilton': defaultdict(<class 'int'>, {'Victor Fleming': 1}), 'Megan Park': defaultdict(<class 'int'>, {'George A. Romero': 1}), 'Teri Garr': defaultdict(<class 'int'>, {'Carroll Ballard': 1}), 'Bhavkhandan Singh Rakhra': defaultdict(<class 'int'>, {'Rohit Jugraj': 1}), 'Iben Hjejle': defaultdict(<class 'int'>, {'Thorbjørn Christoffersen': 1}), 'Rijk de Gooyer': defaultdict(<class 'int'>, {'Alex van Warmerdam': 1}), 'Spencer Tracy': defaultdict(<class 'int'>, {'Victor Fleming': 1}), 'Judie Aronson': defaultdict(<class 'int'>, {'Joseph Zito': 1}), 'Joanna Kulig': defaultdict(<class 'int'>, {'Pawel Pawlikowski': 1}), 'Rupert Graves': defaultdict(<class 'int'>, {'James Ivory': 1}), 'Gerry Lopez': defaultdict(<class 'int'>, {'Stacy Peralta': 1}), 'Karra Elejalde': defaultdict(<class 'int'>, {'Nacho Vigalondo': 1}), 'Jack Kao': defaultdict(<class 'int'>, {'Hao Ning': 1}), 'Brianna Brown': defaultdict(<class 'int'>, {'Tony Giglio': 1}), 'Joe Cross': defaultdict(<class 'int'>, {'Joe Cross': 1}), 'Essence Atkins': defaultdict(<class 'int'>, {'Michael Tiddes': 1, 'David M. Matthews': 1}), 'Barack Obama': defaultdict(<class 'int'>, {'Dinesh D'Souza': 1}), 'Tom Atkins': defaultdict(<class 'int'>, {'Tommy Lee Wallace': 1}), 'Patricia Neal': defaultdict(<class 'int'>, {'Martin Ritt': 1}), 'David Jason Perez': defaultdict(<class 'int'>, {'Leslie Small': 1}), 'Hanno Pöschl': defaultdict(<class 'int'>, {'Richard Linklater': 1}), 'Patrick Huard': defaultdict(<class 'int'>, {'Éric Tessier': 1}), 'Joan Chen': defaultdict(<class 'int'>, {'Alice Wu': 1}), 'Satish Kaushik': defaultdict(<class 'int'>, {'Sarah Gavron': 1}), 'Bill Maher': defaultdict(<class 'int'>, {'Larry Charles': 1}), 'Dustin Lance Black': defaultdict(<class 'int'>, {'Reed Cowan': 1}), 'Benno Färrmann': defaultdict(<class 'int'>, {'Stefan Ruzowitzky': 1}), 'Leonor Varela': defaultdict(<class 'int'>, {'Alex Rivera': 1}), 'Ashley Scott': defaultdict(<class 'int'>, {'John Murlowski': 1}), 'Anthony 'Citric' Campos': defaultdict(<class 'int'>, {'Mike Judge': 1}), 'Simon Merrells': defaultdict(<class 'int'>, {'Zackary Adler': 1}), 'Esther Williams': defaultdict(<class 'int'>, {'George Sidney': 1}), 'Jossara Jinaro': defaultdict(<class 'int'>, {'Carmen Marron': 1}), 'Agnes Moorehead': defaultdict(<class 'int'>, {'George Sidney': 1}), 'Leopold Stokowski': defaultdict(<class 'int'>, {'James Algar': 1}), 'Nicole Smolen': defaultdict(<class 'int'>, {'Jaco Booyens': 1}), 'Richard Brooker': defaultdict(<class 'int'>, {'Steve Miner': 1}), 'Tiffany Helm': defaultdict(<class 'int'>, {'Dany Steinmann': 1}), 'Brian Austin Green': defaultdict(<class 'int'>, {'Hayley Cloake': 1}), 'Jessica Tandy': defaultdict(<class 'int'>, {'Tay Garnett': 1}), 'Saeed Jaffrey': defaultdict(<class 'int'>, {'Harmage Singh Kalirai': 1, 'Stephen Frears': 1}), 'Rachel Skarsten': defaultdict(<class 'int'>, {'Jon Knautz': 1}), 'Myrna Loy': defaultdict(<class 'int'>, {'William Wyler': 1}), 'Jürgen Langhelle': defaultdict(<class 'int'>, {'Petter Næss': 1}), 'Michael Derek': defaultdict(<class 'int'>, {'Robert Fontaine': 1}), 'Manuela Velasco': defaultdict(<class 'int'>, {'Jaume Balaguerá': 1}), 'Travis Myers': defaultdict(<class 'int'>, {'Dan Perr

```

```

i': 1}}, 'Walter Hendrix III': defaultdict(<class 'int'>, {'Neil Mcenery-West': 1}}, 'James Ransone': defaultdict(<class 'int'>, {'Anthony O'Brien': 1}}, 'Phoebe Legere': defaultdict(<class 'int'>, {'Michael Herz': 1}}, 'Carroll O'Connor': defaultdict(<class 'int'>, {'': 1}}, 'Maika Monroe': defaultdict(<class 'int'>, {'David Robert Mitchell': 1}}, 'Vernon Wells': defaultdict(<class 'int'>, {'George Miller': 1}}, 'Alicia Goranson': defaultdict(<class 'int'>, {'Kimberly Peirce': 1}}, 'Eric Sheffer Stevens': defaultdict(<class 'int'>, {'Chris Kentis': 1}}, 'Ben Davies': defaultdict(<class 'int'>, {'Alex Kendrick': 1}}, 'Hedy Lamarr': defaultdict(<class 'int'>, {'Jack Conway': 1}}, 'Michel Robin': defaultdict(<class 'int'>, {'Sylvain Chomet': 1}}, 'Michael Greyeyes': defaultdict(<class 'int'>, {'Chris Eyre': 1}}, 'Vernon Dobtcheff': defaultdict(<class 'int'>, {'Richard Linklater': 1}}, 'Adriana Barraza': defaultdict(<class 'int'>, {'Alejandro G. Iñárritu': 1}}, 'Nicholas Aaron': defaultdict(<class 'int'>, {'Kevin Macdonald': 1}}, 'Tyronne Power': defaultdict(<class 'int'>, {'Henry King': 1}}, 'Najarra Townsend': defaultdict(<class 'int'>, {'Miranda July': 1}}, 'Vanessa Lengies': defaultdict(<class 'int'>, {'Max Joseph': 1}}, 'Henry Silva': defaultdict(<class 'int'>, {'Jim Jarmusch': 1}}, 'Kathleen Wilhoite': defaultdict(<class 'int'>, {'Kevin Tenney': 1}}, 'Stephen Colbert': defaultdict(<class 'int'>, {'': 1}}, 'Sook-Yin Lee': defaultdict(<class 'int'>, {'John Cameron Mitchell': 1}}, 'Ari Folman': defaultdict(<class 'int'>, {'Ari Folman': 1}}, 'Noah Danby': defaultdict(<class 'int'>, {'Gary Rogers': 1}}, 'John F. Kennedy': defaultdict(<class 'int'>, {'David Singleton': 1}}, 'Danny Aiello': defaultdict(<class 'int'>, {'Bob Giraldi': 1}}, 'Alanna Ubach': defaultdict(<class 'int'>, {'Jill Sprecher': 1, 'Rob Bowman': 1}}, 'Matt Bennett': defaultdict(<class 'int'>, {'Huck Botko': 1}}, 'Jess Weixler': defaultdict(<class 'int'>, {'Mitchell Lichtenstein': 1}}, 'Jeffrey Falcon': defaultdict(<class 'int'>, {'Lance Mungia': 1}}, 'Karisma Kapoor': defaultdict(<class 'int'>, {'Kundan Shah': 1, 'Khalid Mohamed': 1}}, 'Paul Kaye': defaultdict(<class 'int'>, {'Michael Dowse': 1}}, 'Merritt Wever': defaultdict(<class 'int'>, {'Noah Buschel': 1}}, 'Brad William Henke': defaultdict(<class 'int'>, {'Laurie Collyer': 1}}, 'Jordi Vilasuso': defaultdict(<class 'int'>, {'Aaron Hann': 1}}, 'Rapulana Seiphe mo': defaultdict(<class 'int'>, {'Ralph Ziman': 1}}, 'Mackenzie Gray': defaultdict(<class 'int'>, {'Colin Minihan': 1}}, 'Peter McQuinn': defaultdict(<class 'int'>, {'Conor McMahon': 1}}, 'Daniel Baldwin': defaultdict(<class 'int'>, {'Chris Shadley': 1}}, 'Alice Cooper': defaultdict(<class 'int'>, {'Simon Napier-Bell': 1}}, 'Paul Sheehan': defaultdict(<class 'int'>, {'Simon Yin': 1}}, 'Steven Weber': defaultdict(<class 'int'>, {'': 1}}, 'Rita Hayworth': defaultdict(<class 'int'>, {'Orson Welles': 1}}, 'Mircea Monroe': defaultdict(<class 'int'>, {'Dave Payne': 1}}, 'Sunil Narkar': defaultdict(<class 'int'>, {'Brad Copeland': 1}}, 'Kate Maberly': defaultdict(<class 'int'>, {'Paul Bunnell': 1}}, 'Jon Mack': defaultdict(<class 'int'>, {'James Cox': 1}}, 'Buster Keaton': defaultdict(<class 'int'>, {'Richard Lester': 1}}, 'Trenton Rostedt': defaultdict(<class 'int'>, {'Scott Dowd': 1}}, 'Melissa Bolona': defaultdict(<class 'int'>, {'Jerry Dugan': 1}}, 'Paul McGillion': defaultdict(<class 'int'>, {'William Kaufman': 1}}, 'Khalid Abdalla': defaultdict(<class 'int'>, {'Jehane Noujaim': 1}}, 'Karl E. Landler': defaultdict(<class 'int'>, {'Pascal Arnold': 1}}, 'Lauren Gottlieb': defaultdict(<class 'int'>, {'Remo': 1}}, 'Paul McGill': defaultdict(<class 'int'>, {'James Marsh': 1}}, 'Siobhan Marshall': defaultdict(<class 'int'>, {'John Laing': 1}}, 'Logan Brownin g': defaultdict(<class 'int'>, {'Jamal Hill': 1}}, 'Caleb Landry Jones': defaultdict(<class 'int'>, {'Daniel Stamm': 1}}, 'Damián Alcázar': defaultdict(<class 'int'>, {'Carlos Carrera': 1}}, 'Donna Murphy': defaultdict(<class 'int'>, {'Vera Farmiga': 1}}, 'Hansika Motwani': defaultdict(<class 'int'>, {'Vijay Chandar': 1}}, 'Tshewang Dendup': defaultdict(<class 'int'>, {'Khyentse Norbu': 1}}, 'Humphrey Bogart': defaultdict(<class 'int'>, {'Richard Brooks': 1, 'Michael Curtiz': 1}}, 'George Newbern': defaultdict(<class 'int'>, {'Tony Krantz': 1}}, 'Kimberley Crossman': defaultdict(<class 'int'>, {'Mitchell Altieri': 1}}, 'Christian Keyes': defaultdict(<class 'int'>, {'Corey Grant': 1}}, 'James Duval': defaultdict(<class 'int'>, {'Lucky McKee': 1}}, 'Matthew Alan': defaultdict(<class 'int'>, {'David Hunt': 1}}, 'Sarah Butler': defaultdict(<class 'int'>, {'Steven R. Monroe': 1}}, 'Ally Walker': defaultdict(<class 'int'>, {'Mark Illsley': 1}}, 'Nawazuddin Siddiqui': defaultdict(<class 'int'>, {'Ritesh Batra': 1}}, 'Jean Arthur': default

```

```

tddict(<class 'int'>, {'Frank Capra': 1}), 'Harry Carey Jr.': defaultdict(<class
'int'>, {'John Ford': 1}), 'Shawnee Smith': defaultdict(<class 'int'>, {'Brad J.
Silverman': 1}), 'Yasmine Al Massri': defaultdict(<class 'int'>, {'Nadine Labak
i': 1}), 'Ohad Knoller': defaultdict(<class 'int'>, {'Eytan Fox': 1}), 'Annabelle
Wallis': defaultdict(<class 'int'>, {'Ramesh Sharma': 1}), 'Dustin Ingram': defau
ltdict(<class 'int'>, {'Travis Zariwny': 1}), 'Lew Temple': defaultdict(<class 'i
nt'>, {'Adrienne Shelly': 1}), 'William Baldwin': defaultdict(<class 'int'>, {'No
ah Baumbach': 1}), 'Scott Cohen': defaultdict(<class 'int'>, {'Charles Herman-Wur
mfeld': 1}), 'Matthew Ziff': defaultdict(<class 'int'>, {'John Stockwell': 1, 'Jo
hnnny Remo': 1, 'Travis Romero': 1}), 'Norman Lloyd': defaultdict(<class 'int'>,
{'Alfred Hitchcock': 1}), 'Ami Ayalon': defaultdict(<class 'int'>, {'Dror Moreh':
1}), 'Beau Bridges': defaultdict(<class 'int'>, {'Rebecca Miller': 1}), 'Donald G
lover': defaultdict(<class 'int'>, {'Maggie Carey': 1}), 'Ward G. Smith': default
dict(<class 'int'>, {'Simeon Rice': 1}), 'Paul Guilfoyle': defaultdict(<class 'in
t'>, {'Brad Anderson': 1}), 'Jessy Schram': defaultdict(<class 'int'>, {'Jeff Gar
lin': 1}), 'Joel David Moore': defaultdict(<class 'int'>, {'Adam Green': 1}), 'Pa
ulette Goddard': defaultdict(<class 'int'>, {'Charles Chaplin': 1}), 'Jay Mohr':
defaultdict(<class 'int'>, {'Adam Carolla': 1}), 'Brian Lee Franklin': defaultdic
t(<class 'int'>, {'John Sayles': 1}), 'Angela Sarafyan': defaultdict(<class 'in
t'>, {'Jaime Zevallos': 1}), "Morgana O'Reilly": defaultdict(<class 'int'>, {'Ger
ard Johnstone': 1}), 'Lee Scott': defaultdict(<class 'int'>, {'Robert Greenwald':
1}), 'Nicole Muñoz': defaultdict(<class 'int'>, {'David Ray': 1}), 'Monica Barla
deanu': defaultdict(<class 'int'>, {'Nae Caranfil': 1}), 'Ashanti': defaultdict(<
class 'int'>, {'David Winning': 1}), 'Brighton Sharbino': defaultdict(<class 'in
t'>, {'Frank Lotito': 1, 'E.L. Katz': 1}), 'Katrina Law': defaultdict(<class 'in
t'>, {'Timothy Woodward Jr.': 1}), 'Timothy Hutton': defaultdict(<class 'int'>,
{'Tara Subkoff': 1}), 'Adriana Caselotti': defaultdict(<class 'int'>, {'William C
ottrell': 1}), 'Brigitte Bardot': defaultdict(<class 'int'>, {'Edward Dmytryk':
1}), 'Zak Penn': defaultdict(<class 'int'>, {'Zak Penn': 1}), 'Camren Bicondova':
defaultdict(<class 'int'>, {'Jon Knautz': 1}), 'Jessica Szohr': defaultdict(<clas
s 'int'>, {'David Worth': 1}), 'Burgess Meredith': defaultdict(<class 'int'>, {'L
eslie H. Martinson': 1}), 'Karl Malden': defaultdict(<class 'int'>, {'': 1}), 'Ul
rich Thomsen': defaultdict(<class 'int'>, {'Thomas Vinterberg': 1}), 'Long Nguye
n': defaultdict(<class 'int'>, {'Ham Tran': 1}), 'James Morrison': defaultdict(<c
lass 'int'>, {'': 1}), 'Elina Abai Kyzy': defaultdict(<class 'int'>, {'Sadyk Sher
-Niyaz': 1}), 'Maury Chaykin': defaultdict(<class 'int'>, {'Paul Donovan': 1}),
'Noureen DeWulf': defaultdict(<class 'int'>, {'Craig Moss': 1}), 'Betsy Palmer':
defaultdict(<class 'int'>, {'Steve Miner': 1}), 'Ray Milland': defaultdict(<class
'int'>, {'Billy Wilder': 1}), 'Vic Mignogna': defaultdict(<class 'int'>, {'Deryck
Broom': 1}), 'Geoffrey Arend': defaultdict(<class 'int'>, {'Jay Chandrasekhar':
1}), 'Zuhair Haddad': defaultdict(<class 'int'>, {'Giovanni Zelko': 1}), 'Ximena
Ayala': defaultdict(<class 'int'>, {'Claudia Sainte-Luce': 1}), 'Jerry Mathers':
defaultdict(<class 'int'>, {'Alfred Hitchcock': 1}), 'Spring Byington': defaulttdi
ct(<class 'int'>, {'Walter Lang': 1}), 'Alex Huszar': defaultdict(<class 'int'>,
{'Anna Muylaert': 1}), 'Jason Marsden': defaultdict(<class 'int'>, {'Steve Taylo
r': 1}), 'Lemmy': defaultdict(<class 'int'>, {'Kurt Voss': 1}), 'DÃ©borah RÃ©vy':
defaultdict(<class 'int'>, {'Laurent Bouhnik': 1}), 'Nataniel SÃ¡nchez': defaultd
ict(<class 'int'>, {'Tom Sanchez': 1}), 'Rich Ceraulo': defaultdict(<class 'in
t'>, {'Adam Jay Epstein': 1}), 'Errol Flynn': defaultdict(<class 'int'>, {'Michae
l Curtiz': 1}), 'Christie Burke': defaultdict(<class 'int'>, {'Brenton Spencer':
1}), 'Redd Foxx': defaultdict(<class 'int'>, {'Ossie Davis': 1}), "Seregon O'Dass
ey": defaultdict(<class 'int'>, {'Dylan Bank': 1}), 'Richard Cawthorne': defaulttd
ict(<class 'int'>, {'': 1}), 'Terry Kiser': defaultdict(<class 'int'>, {'Jeff Bur
r': 1}), 'Sarah Murphy-Dyson': defaultdict(<class 'int'>, {'Guy Maddin': 1}), 'Se
an Cameron Michael': defaultdict(<class 'int'>, {'Regardt van den Bergh': 1}), 'C
hris Gauthier': defaultdict(<class 'int'>, {'Panos Cosmatos': 1}), 'Iko Uwais': d
efaultdict(<class 'int'>, {'Gareth Evans': 1}), 'Piero MaggiÃ²': defaultdict(<cla
ss 'int'>, {'Joe Marino': 1}), 'Talia Zucker': defaultdict(<class 'int'>, {'Joel
Anderson': 1}), 'Christopher Abbott': defaultdict(<class 'int'>, {'Mona Fastvol
d': 1, 'Sean Durkin': 1}), 'Shelley Hennig': defaultdict(<class 'int'>, {'Levan G

```

```

abriadze': 1}}, 'Ursula Andress': defaultdict(<class 'int'>, {'Terence Young': 1}}, 'Andrew Robinson': defaultdict(<class 'int'>, {'Clive Barker': 1}}, 'Benjamin A. Onyango': defaultdict(<class 'int'>, {'Harold Cronk': 1}}, 'Hiroshi Abe': defaultdict(<class 'int'>, {'Takao Okawara': 1}}, 'Fernanda Andrade': defaultdict(<class 'int'>, {'William Brent Bell': 1}}, 'James Drury': defaultdict(<class 'int'>, {'Robert D. Webb': 1}}, 'Billy West': defaultdict(<class 'int'>, {'Davis Guggenheim': 1}}, 'James Brown': defaultdict(<class 'int'>, {'Allan Dwan': 1}}, 'Mick Jagger': defaultdict(<class 'int'>, {'Martin Scorsese': 1}}, 'Juliet Stevenson': defaultdict(<class 'int'>, {'Andrew Steggall': 1}}, 'W. Earl Brown': defaultdict(<class 'int'>, {'Ben Lewin': 1}}, 'Michael Pollan': defaultdict(<class 'int'>, {'Robert Kenner': 1}}, 'Gaby Hoffmann': defaultdict(<class 'int'>, {'Gillian Robespierre': 1}}, 'Sheryl Crow': defaultdict(<class 'int'>, {'Morgan Neville': 1}}, 'Kevin Corrigan': defaultdict(<class 'int'>, {'Nicole Holofcener': 1, 'Steve Buscemi': 1, 'Jack Perez': 1}}, 'Sonja Sohn': defaultdict(<class 'int'>, {'Marc Levin': 1}}, 'Trey Parker': defaultdict(<class 'int'>, {'Trey Parker': 1}}, 'Shahir Kabaha': defaultdict(<class 'int'>, {'Scandar Copti': 1}}, 'Steven Yeun': defaultdict(<class 'int'>, {'Mike Cahill': 1}}, 'Shiri Appleby': defaultdict(<class 'int'>, {'Efram Potelle': 1}}, 'Anwar Congo': defaultdict(<class 'int'>, {'Joshua Oppenheimer': 1}}, 'Alex Gibney': defaultdict(<class 'int'>, {'Alex Gibney': 1}}, 'PelÃ': defaultdict(<class 'int'>, {'Paul Crowder': 1}}, 'Kumar Gaurav': defaultdict(<class 'int'>, {'Rohit Jagessar': 1}}, 'Josh Swanson': defaultdict(<class 'int'>, {'Anthony Powell': 1}}, 'G.W. Krauss': defaultdict(<class 'int'>, {'Kief Davidson': 1}}, 'Emraan Hashmi': defaultdict(<class 'int'>, {'Vivek Agnihotri': 1}}, 'Lena Dunham': defaultdict(<class 'int'>, {'Ti West': 2, 'Daniel Schechter': 1, 'Lena Dunham': 1}}, 'Harvey Fierstein': defaultdict(<class 'int'>, {'John Hamburg': 1}}, 'Patrick Renna': defaultdict(<class 'int'>, {'Nickolas Perry': 1}}, 'Yul Vazquez': defaultdict(<class 'int'>, {'Dan Zukovic': 1}}, 'Will Rothhaar': defaultdict(<class 'int'>, {'Becky Smith': 1}}, 'Jordana Spiro': defaultdict(<class 'int'>, {'Eric Nicholas': 1}}, 'Nora Zehetner': defaultdict(<class 'int'>, {'Benjamin Dickinson': 1}}, 'Frances Bay': defaultdict(<class 'int'>, {'Darren Stein': 1}}, 'Scott Levy': defaultdict(<class 'int'>, {'A. Raven Cruz': 1}}, 'Kenny Johnston': defaultdict(<class 'int'>, {'Amat Escalante': 1}}, 'Rasheed Naz': defaultdict(<class 'int'>, {'Wajahat Rauf': 1}}, 'Ilene Graff': defaultdict(<class 'int'>, {'Katherine Brooks': 1}}, 'Jason Ritter': defaultdict(<class 'int'>, {'David Cross': 1}}, 'Nicole Randall Johnson': defaultdict(<class 'int'>, {'Randall Rubin': 1}}, 'Randy Wayne': defaultdict(<class 'int'>, {'Robert Bennett': 1, 'Brian Bugh': 1}}, 'Russell Hodgkinson': defaultdict(<class 'int'>, {'Kevin Hamedani': 1}}, 'Maureen McCormick': defaultdict(<class 'int'>, {'David DeCoteau': 1}}, 'Ãx81lex Angulo': defaultdict(<class 'int'>, {'Mikel Rueda': 1}}, 'Katie Aselton': defaultdict(<class 'int'>, {'Katie Aselton': 1}}, 'David Oakes': defaultdict(<class 'int'>, {'Robert Heath': 1}}, 'Magi Avila': defaultdict(<class 'int'>, {'D. Stevens': 1}}, 'Boyd Banks': defaultdict(<class 'int'>, {'Jeffrey St. Jules': 1}}, 'Angelina Armani': defaultdict(<class 'int'>, {'Robert Hall': 1}}, 'Jason K. Wixom': defaultdict(<class 'int'>, {'K. King': 1}}, 'Justin Baldoni': defaultdict(<class 'int'>, {'Jonathan Meyers': 1}}, 'Monique Candelaria': defaultdict(<class 'int'>, {'Blair Erickson': 1}}, 'Eric Lloyd': defaultdict(<class 'int'>, {'': 1}}, 'Kimberly Williams-Paisley': defaultdict(<class 'int'>, {'Allison Burnett': 1}}, 'Alma Martinez': defaultdict(<class 'int'>, {'Sanjay Rawal': 1}}, 'Justin Kirk': defaultdict(<class 'int'>, {'Kat Coiro': 1}}, 'Anamaria Marinca': defaultdict(<class 'int'>, {'Cristian Mungiu': 1}}, 'Odessa Rae': defaultdict(<class 'int'>, {'David Slade': 1}}, 'Sarah Kazemy': defaultdict(<class 'int'>, {'Maryam Keshavarz': 1}}, 'Ahna O'Reilly': defaultdict(<class 'int'>, {'Ryan Coogler': 1}}, 'Sam Edwards': defaultdict(<class 'int'>, {'James Algar': 1}}, 'John Enos III': defaultdict(<class 'int'>, {'Charles Herman-Wurmfeld': 1}}, 'Stana Roumillac': defaultdict(<class 'int'>, {'Marianne Monpierre': 1}}, 'James Wolk': defaultdict(<class 'int'>, {'Jamie Travis': 1}}, 'Gavin MacLeod': defaultdict(<class 'int'>, {'Rich Christiano': 1}}, 'Lorraine Stanley': defaultdict(<class 'int'>, {'Paul Andrew Williams': 1}}, 'Tommy O'Reilly': defaultdict(<class 'int'>, {'Natalie Bible': 1}}, 'Shahab Hossaini': defaultdict(<class 'int'>, {'Asghar Farhadi': 1}}, 'Jo Hartley': defaultdict(<class 'int'>, {'Justin Molotnikov': 1}}, 'Allison Dean': defaultdict(<class

```



```

'int'>, {'Victor Nunez': 1}}, 'Victor Rasuk': defaultdict(<class 'int'>, {'Peter
Sollett': 1}), 'Louise Brooks': defaultdict(<class 'int'>, {'Georg Wilhelm Pabs
t': 1}), 'Norma Aleandro': defaultdict(<class 'int'>, {'Jorge Gaggero': 1}), 'Zac
hary Knighton': defaultdict(<class 'int'>, {'Michael Burke': 1}), 'Shane Dawson':
defaultdict(<class 'int'>, {'Shane Dawson': 1}), 'Corbin Allred': defaultdict(<cl
ass 'int'>, {'Ryan Little': 1}), 'Taylor Pigeon': defaultdict(<class 'int'>, {'Cl
ark Baker': 1}), 'Scott Helvenston': defaultdict(<class 'int'>, {'Robert Greenwal
d': 1}), 'Isaac C. Singleton Jr.': defaultdict(<class 'int'>, {'Leslie Small':
1}), 'Jean-Louis Trintignant': defaultdict(<class 'int'>, {'Bernardo Bertolucci':
1}), 'Raymond J. Barry': defaultdict(<class 'int'>, {'Neil Burger': 1}), 'Luke Gr
imes': defaultdict(<class 'int'>, {'Jonathan Levine': 1}), 'Dichen Lachman': defa
ultdict(<class 'int'>, {'Christopher Hutson': 1}), 'William Gates': defaultdict(<
class 'int'>, {'Steve James': 1}), 'Kevin Scott Richardson': defaultdict(<class
'int'>, {'David LaChapelle': 1}), 'Adam LeFevre': defaultdict(<class 'int'>, {'Mi
chael Cuesta': 1}), 'Laura Fraser': defaultdict(<class 'int'>, {'Caryn Waechter':
1}), 'Irene Bedard': defaultdict(<class 'int'>, {'Jennifer Wynne Farmer': 1}), 'H
ill Harper': defaultdict(<class 'int'>, {'Christopher Scott Cherot': 1}), 'Trey L
oney': defaultdict(<class 'int'>, {'Randy Moore': 1}), 'Richard Curtis': defaultt
dict(<class 'int'>, {'Chris Atkins': 1}), 'Leven Rambin': defaultdict(<class 'in
t'>, {'Anna Mastro': 1}), 'Ginger Rogers': defaultdict(<class 'int'>, {'Mark Sand
rich': 1, 'Lloyd Bacon': 1}), 'Heather Donahue': defaultdict(<class 'int'>, {'Dan
iel Myrick': 1}), 'Joe Cocker': defaultdict(<class 'int'>, {'Michael Wadleigh':
1}), 'David Zucker': defaultdict(<class 'int'>, {'John Landis': 1}), 'Stacy Keac
h': defaultdict(<class 'int'>, {'Jon Gunn': 1, 'Matt Cimber': 1}), 'Kendyl Joi':
defaultdict(<class 'int'>, {'Gene Teigland': 1}), 'Aliou TourÃ©': defaultdict(<cl
ass 'int'>, {'Johanna Schwartz': 1}), 'Karen Alexander': defaultdict(<class 'in
t'>, {'William Gazecki': 1}), 'Russell Wong': defaultdict(<class 'int'>, {'Lance
McDaniel': 1}), 'Paula GarcÃa': defaultdict(<class 'int'>, {'Michael Walker':
1}), 'Paul McCartney': defaultdict(<class 'int'>, {'Richard Lester': 1}), 'Dario
Nozic Serini': defaultdict(<class 'int'>, {'Blaz Završnik': 1}), 'Kirk Cameron':
defaultdict(<class 'int'>, {'Alex Kendrick': 1}), 'Bhole Baba': defaultdict(<clas
s 'int'>, {'Pan Nalin': 1}), 'Frances Bavier': defaultdict(<class 'int'>, {'Joe C
amp': 1}), 'Blanchard Ryan': defaultdict(<class 'int'>, {'Chris Kentis': 1}), 'Wo
ody Strode': defaultdict(<class 'int'>, {'John 'Bud' Cardos': 1}), 'Heather Berma
n': defaultdict(<class 'int'>, {'Marilyn Agrelo': 1}), 'Ken Burns': defaultdict(<
class 'int'>, {'Patrick Creadon': 1}), 'Terry Funk': defaultdict(<class 'int'>,
{'Barry W. Blaustein': 1}), 'Kirby Heyborne': defaultdict(<class 'int'>, {'Kurt H
ale': 2}), 'Marina Golbahari': defaultdict(<class 'int'>, {'Siddiq Barmak': 1}),
'Rachel Dratch': defaultdict(<class 'int'>, {'Joseph Dorman': 1}), 'Rachel True':
defaultdict(<class 'int'>, {'Greg Harrison': 1}), 'Sasha Alexander': defaultdict
(<class 'int'>, {'Michael Polish': 1}), 'Edie Falco': defaultdict(<class 'int'>,
{'Morgan J. Freeman': 1}), 'Monica Calhoun': defaultdict(<class 'int'>, {'Neema B
arnette': 1}), 'David Mamet': defaultdict(<class 'int'>, {'Molly Bernstein': 1}),
'Takashi Shimura': defaultdict(<class 'int'>, {'Akira Kurosawa': 1}), 'Tommy Shep
pard': defaultdict(<class 'int'>, {'Marius A. Markevicius': 1}), 'Monique Gabriel
a Curnen': defaultdict(<class 'int'>, {'Justin Lin': 1}), 'Haley Ramm': defaulttdi
ct(<class 'int'>, {'Quentin Dupieux': 1}), 'Tori Spelling': defaultdict(<class 'i
nt'>, {'C. Jay Cox': 1}), 'Bo Huang': defaultdict(<class 'int'>, {'Hao Ning':
1}), 'Brian Eno': defaultdict(<class 'int'>, {'Stephen Kijak': 1}), 'Daston Kalil
i': defaultdict(<class 'int'>, {'Daston Kalili': 1}), 'Carol Channing': defaulttdi
ct(<class 'int'>, {'William Gazecki': 1}), 'Vincent Chimato': defaultdict(<class
'int'>, {'Michael Hoffman Jr.': 1}), 'Emma Caulfield': defaultdict(<class 'int'>,
{'Patrick Gilles': 1}), 'Mike Beckingham': defaultdict(<class 'int'>, {'Georgia H
ilton': 1}), 'Shaun Sipos': defaultdict(<class 'int'>, {'Jack Heller': 1}), 'Mann
y Perez': defaultdict(<class 'int'>, {'Fernando Baez Mella': 1}), 'Jennifer Hal
e': defaultdict(<class 'int'>, {'Warren Sheppard': 1}), 'Mary Kate Wiles': defaul
tdict(<class 'int'>, {'Justin Paul Miller': 1}), 'Carrie Bradstreet': defaultdict
(<class 'int'>, {'Joel Paul Reisig': 1}), 'Elizabeth Streb': defaultdict(<class
'int'>, {'Catherine Gund': 1}), 'Mike Stanley': defaultdict(<class 'int'>, {'Luke
Dye': 1}), 'Patrick Kilpatrick': defaultdict(<class 'int'>, {'Michael Herz': 1}),

```

```

'Lawrence Gilliard Jr.': defaultdict(<class 'int'>, {'Matty Rich': 1}), 'Luis Sanchez': defaultdict(<class 'int'>, {'Alec Astin': 1}), 'John Karyus': defaultdict(<class 'int'>, {'Lloyd Kaufman': 1}), 'Jennifer Lafleur': defaultdict(<class 'int'>, {'Matthew Watts': 1}), 'Bob Proctor': defaultdict(<class 'int'>, {'Joe Kenemore': 1}), 'Chris Eigeman': defaultdict(<class 'int'>, {'Whit Stillman': 1}), 'Jack E. Currenton': defaultdict(<class 'int'>, {'Eric England': 1}), 'Micah Sloat': defaultdict(<class 'int'>, {'Oren Peli': 1}), 'Tony Alva': defaultdict(<class 'int'>, {'Stacy Peralta': 1}), 'Emily Rios': defaultdict(<class 'int'>, {'Richard Glatzer': 1}), 'Tony Doupe': defaultdict(<class 'int'>, {'Sue Corcoran': 1}), 'Greg Ayres': defaultdict(<class 'int'>, {'Jonathan Caouette': 1}), 'Bill Farmer': defaultdict(<class 'int'>, {'Ray Griggs': 1}), 'Catriona MacColl': defaultdict(<class 'int'>, {'Lucio Fulci': 1}), 'Brendon Small': defaultdict(<class 'int'>, {'': 1}), 'Treat Williams': defaultdict(<class 'int'>, {'Tom Vaughan': 1}), 'Dov Tiefenbach': defaultdict(<class 'int'>, {'Paul Fox': 1}), 'R. Brandon Johnson': defaultdict(<class 'int'>, {'Ari Kirschenbaum': 1, 'Stevan Mena': 1}), 'Liv Ullmann': defaultdict(<class 'int'>, {'Ingmar Bergman': 1, 'Laslo Benedek': 1}), 'Lillian Gish': defaultdict(<class 'int'>, {'D.W. Griffith': 1}), 'Walter Koenig': defaultdict(<class 'int'>, {'Roger Nygard': 1}), 'Anita Page': defaultdict(<class 'int'>, {'Harry Beaumont': 1}), 'America Olivo': defaultdict(<class 'int'>, {'Franck Khalifoun': 1}), 'Amos Oz': defaultdict(<class 'int'>, {'Mor Loushy': 1}), 'Mark Zupan': defaultdict(<class 'int'>, {'Henry Alex Rubin': 1}), 'Michael Dudikoff': defaultdict(<class 'int'>, {'Sam Firstenberg': 1}), 'Carol Block': defaultdict(<class 'int'>, {'Doug Block': 1}), 'Tom E. Nicholson': defaultdict(<class 'int'>, {'Chad Kapper': 1}), 'Jack Warden': defaultdict(<class 'int'>, {'Sidney Lumet': 1}), 'Claudette Colbert': defaultdict(<class 'int'>, {'Frank Capra': 1}), 'Angeliki Papoulia': defaultdict(<class 'int'>, {'Yorgos Lanthimos': 1}), 'James Cagney': defaultdict(<class 'int'>, {'Lauren Lazin': 1}), 'Kimberly J. Brown': defaultdict(<class 'int'>, {'Gavin O'Connor': 1}), 'Simon Abkarian': defaultdict(<class 'int'>, {'Cădric Klapisch': 1}), 'Kevin Howarth': defaultdict(<class 'int'>, {'Niall Johnson': 1}), 'Daniel London': defaultdict(<class 'int'>, {'Kelly Reichardt': 1}), 'John Robinson': defaultdict(<class 'int'>, {'Kelly Reichardt': 1}), 'Embeth Davitz': defaultdict(<class 'int'>, {'Eric Mendelsohn': 1}), 'Jean-Paul Belmondo': defaultdict(<class 'int'>, {'Jean-Luc Godard': 1}), 'Vera Ngassa': defaultdict(<class 'int'>, {'Florence Ayisi': 1}), 'Robert John Burke': defaultdict(<class 'int'>, {'Jesse Peretz': 1}), 'Don Wallace': defaultdict(<class 'int'>, {'Eddie O'Flaherty': 1}), 'Lalaine': defaultdict(<class 'int'>, {'Babar Ahmed': 1}), 'Trish Basinger': defaultdict(<class 'int'>, {'John D. Hancock': 1}), 'Tatyana Ali': defaultdict(<class 'int'>, {'Bruce Dellis': 1}), 'Michael Des Barres': defaultdict(<class 'int'>, {'Pece Dingo': 1}), 'Rumi Missabu': defaultdict(<class 'int'>, {'Åx89tienne Faure': 1}), 'Dreema Walker': defaultdict(<class 'int'>, {'Craig Zobel': 1}), 'Aunjanue Ellis': defaultdict(<class 'int'>, {'Nicole Holofcener': 1}), 'Hector Echavarria': defaultdict(<class 'int'>, {'Ken Del Conte': 1}), 'Parry Shen': defaultdict(<class 'int'>, {'Justin Lin': 1}), 'Nicole Ari Parker': defaultdict(<class 'int'>, {'Maria Maggenti': 1}), 'Purva Bedi': defaultdict(<class 'int'>, {'Piyush Dinker Pandya': 1}), 'Keri Maletto': defaultdict(<class 'int'>, {'Daniel Columbie': 1}), 'Charis Michelsen': defaultdict(<class 'int'>, {'Bill Plympton': 1}), 'Amir Talai': defaultdict(<class 'int'>, {'Greg Harrison': 1}), 'Marshal Hilton': defaultdict(<class 'int'>, {'Al Franklin': 1}), 'Ally Sheedy': defaultdict(<class 'int'>, {'Allison Anders': 1}), 'Samantha Futerman': defaultdict(<class 'int'>, {'Michael Kang': 1}), 'Nolan Gerard Funk': defaultdict(<class 'int'>, {'Paul Schrader': 1}), 'Judy Marte': defaultdict(<class 'int'>, {'Lori Silverbush': 1}), 'Michael Abbott Jr.': defaultdict(<class 'int'>, {'Jeff Nichols': 1}), 'James Urbaniak': defaultdict(<class 'int'>, {'Hilary Brougher': 1}), 'Dustin Fitzsimons': defaultdict(<class 'int'>, {'Rachel Goldenberg': 1}), 'Eme Ikwaakor': defaultdict(<class 'int'>, {'Jamin Winans': 1}), 'Florence Henderson': defaultdict(<class 'int'>, {'Tom Seidman': 1}), 'Victoria Jackson': defaultdict(<class 'int'>, {'Jason Naumann': 1}), 'Sam Adams': defaultdict(<class 'int'>, {'Sam Martin': 1}), 'Noah Segal': defaultdict(<class 'int'>, {'Andrew Hyatt': 1}), 'Joshua Ray Bell': defaultdict(<class 'int'>, {'Joel Paul Reisig': 1}), 'John Gilbert': defaultdict(<class 'int'>, {'King Vidor': 1}), 'Matthew Emerick': defaultdict(<class 'i

```

```

nt'}, {'Zoran Lisinac': 1}}, 'Mark Fantasia': defaultdict(<class 'int'>, {'James
O'Brien': 1}}, 'Brendan Doogie Milewski': defaultdict(<class 'int'>, {'Tom Putna
m': 1}}, 'Dan Futterman': defaultdict(<class 'int'>, {'Jon Shear': 1}}, 'Christin
a Hart': defaultdict(<class 'int'>, {'Al Silliman Jr.': 1}}, 'Kenneth Tobey': def
aultdict(<class 'int'>, {'EugÃ¨ne LouriÃ©': 1}}, 'Hugh Keays-Byrne': defaultdict
(<class 'int'>, {'George Miller': 1}}, 'Mae West': defaultdict(<class 'int'>, {'L
owell Sherman': 1}}, 'The Dalai Lama': defaultdict(<class 'int'>, {'Maurizio Bena
zzo': 1}}, 'Michael Joiner': defaultdict(<class 'int'>, {'David G. Evans': 1}},
'Devid Striesow': defaultdict(<class 'int'>, {'Tom Tykwer': 1}}, 'William Joseph
Elk III': defaultdict(<class 'int'>, {'Sherman Alexie': 1}}, 'Matisyahu': default
dict(<class 'int'>, {'Justin Dillon': 1}}, 'Marri Savinar': defaultdict(<class 'i
nt'>, {'Adam Rifkin': 1}}, 'Melanie Papalia': defaultdict(<class 'int'>, {'Eric V
alette': 1}}, 'Susan Chambers': defaultdict(<class 'int'>, {'Livingston Oden':
1}}, 'Marina Vlady': defaultdict(<class 'int'>, {'Chris Marker': 1}}, 'Hanne Age
sen': defaultdict(<class 'int'>, {'Carl Theodor Dreyer': 1}}, 'William Katt': def
aultdict(<class 'int'>, {'Richard Schenkman': 1}}, 'Darryl Hunt': defaultdict(<cl
ass 'int'>, {'Ricki Stern': 1}}, 'John Newton': defaultdict(<class 'int'>, {'Jame
s Kerwin': 1}}, 'Elaine Bromka': defaultdict(<class 'int'>, {'C. Fraser Press':
1}}, 'Paul Hickert': defaultdict(<class 'int'>, {'Rania Attieh': 1}}, 'Bahare Sed
idqi': defaultdict(<class 'int'>, {'Majid Majidi': 1}}, 'S. Epatha Merkerson': de
faultdict(<class 'int'>, {'Spike Lee': 1}}, 'Abigail Evans': defaultdict(<class
'int'>, {'Cary Bell': 1}}, 'Iulia Ciochina': defaultdict(<class 'int'>, {'Nicolae
Constantin Tanase': 1}}, 'Robin Lord Taylor': defaultdict(<class 'int'>, {'Mike C
ahill': 1}}, 'Charla Cochran': defaultdict(<class 'int'>, {'Ken Roht': 1}}, 'Bebe
Neuwirth': defaultdict(<class 'int'>, {'Gary Winick': 1}}, 'Glen Hansard': default
tdict(<class 'int'>, {'John Carney': 1}}, 'Marilyn Rising': defaultdict(<class 'i
nt'>, {'Robinson Devor': 1}}, 'Temple Grandin': defaultdict(<class 'int'>, {'Mich
el Orion Scott': 1}}, 'Alan Brent': defaultdict(<class 'int'>, {'Pat Holden':
1}}, 'Joel Moody': defaultdict(<class 'int'>, {'Eric Bugbee': 1}}, 'Peter Robbin
s': defaultdict(<class 'int'>, {'Bill Melendez': 1}}, 'Naderev Sano': defaultdict
(<class 'int'>, {'Dena Seidel': 1}}, 'Lisa Ann': defaultdict(<class 'int'>, {'Deb
orah Anderson': 1}}, 'Ariel Hsing': defaultdict(<class 'int'>, {'Sara Newens':
1}}, 'J.D. Williams': defaultdict(<class 'int'>, {'Sai Varadan': 1}}, 'Christophe
r Denham': defaultdict(<class 'int'>, {'Zal Batmanglij': 1}}, 'Mark Duplass': def
aultdict(<class 'int'>, {'Lynn Shelton': 1, 'Jay Duplass': 1}}, 'Diane Sorrentin
o': defaultdict(<class 'int'>, {'Valentine': 1}}, 'Philippe Leroy': defaultdict(<
class 'int'>, {'Jean-Luc Godard': 1}}, 'Jason Stuart': defaultdict(<class 'int'>,
{'Nate Parker': 1}}, 'Richard Moll': defaultdict(<class 'int'>, {'Nathan Smith Jo
nes': 1}}, 'Pfeifer Brown': defaultdict(<class 'int'>, {'Travis Cluff': 1}}, 'Hal
Landon Jr.': defaultdict(<class 'int'>, {'David Lynch': 1}}, 'Stephen Carr': defa
ultdict(<class 'int'>, {'Harry F. Millarde': 1}}, 'Robert Townsend': defaultdict
(<class 'int'>, {'Robert Townsend': 1}}, 'Leon Isaac Kennedy': defaultdict(<class
'int'>, {'Jamaa Fanaka': 1}}, 'Fay Masterson': defaultdict(<class 'int'>, {'Larry
Blamire': 1}}, 'Kevin P. Farley': defaultdict(<class 'int'>, {'Stephen Langford':
1}}, 'Jonathan Blow': defaultdict(<class 'int'>, {'Lisanne Pajot': 1}}, 'Sheldon
Leonard': defaultdict(<class 'int'>, {'John Reinhardt': 1}}, 'Claire Gordon-Harpe
r': defaultdict(<class 'int'>, {'Patrick Ryan Sims': 1}}, 'Jon Brion': defaultdic
t(<class 'int'>, {'Jason Miller': 1}}, 'Jack McGee': defaultdict(<class 'int'>,
{'Julie Davis': 1}}, 'Michelle Simone Miller': defaultdict(<class 'int'>, {'Brett
Piper': 1}}, 'Chemeeka Walker': defaultdict(<class 'int'>, {'Morgan Spurlock':
1}}, 'Virginia Leith': defaultdict(<class 'int'>, {'Joseph Green': 1}}, 'Ty Parke
r': defaultdict(<class 'int'>, {'Wade Gasque': 1}}, 'Justin Gordon': defaultdict
(<class 'int'>, {'Mike Flanagan': 1}}, 'Shari Albert': defaultdict(<class 'int'>,
{'Edward Burns': 1}}, 'Paul Daniel Ayotte': defaultdict(<class 'int'>, {'Matt Joh
nson': 1}}, 'Marcello Mastroianni': defaultdict(<class 'int'>, {'Bruno Barreto':
1}}, 'Dan Novy': defaultdict(<class 'int'>, {'Andrew Leman': 1}}, 'Chris 'Wonder'
Schoeck': defaultdict(<class 'int'>, {'Dave Carroll': 1}}, 'Taliesin Jaffe': defa
ultdict(<class 'int'>, {'Patrick Meaney': 1}}, 'Jan Haley': defaultdict(<class 'i
nt'>, {'Chad Hartigan': 1}}, 'Paul Schneider': defaultdict(<class 'int'>, {'David
Gordon Green': 1}}, 'Derick Martini': defaultdict(<class 'int'>, {'Kevin Jordan':

```

```
1})), 'Elissa Dowling': defaultdict(<class 'int'>, {'Travis Legge': 1}), 'Ringo St
arr': defaultdict(<class 'int'>, {'Martin Scorsese': 1}), 'Julianne Gabert': defa
ultdict(<class 'int'>, {'Collin Joseph Neal': 1}), 'Joseph Campanella': defaultdi
ct(<class 'int'>, {'Mike Bruce': 1}), 'Andrew Bujalski': defaultdict(<class 'in
t'>, {'Andrew Bujalski': 2}), 'Nichole Ceballos': defaultdict(<class 'int'>, {'Da
mir Catic': 1}), 'Don Brooks': defaultdict(<class 'int'>, {'James Bidgood': 1}),
'Stacy Edwards': defaultdict(<class 'int'>, {'Neil LaBute': 1}), 'Franky G': defa
ultdict(<class 'int'>, {'Eric Eason': 1}), 'Tommy Pallotta': defaultdict(<class
'int'>, {'Richard Linklater': 1}), 'Tjasa Ferme': defaultdict(<class 'int'>, {'Jo
seph Mazzella': 1}), 'Kristen Seavey': defaultdict(<class 'int'>, {'Travis Legg
e': 1}), 'Shannen Fields': defaultdict(<class 'int'>, {'Alex Kendrick': 1}), 'Ash
ley Tramonte': defaultdict(<class 'int'>, {'Marcus Nispel': 1}), 'Robbie Barnes':
defaultdict(<class 'int'>, {'Brandon Landers': 1}), 'Paul Ogola': defaultdict(<cl
ass 'int'>, {'Jim Chuchu': 1}), 'Zoe Lister-Jones': defaultdict(<class 'int'>,
{'Daryl Wein': 1}), 'Sean Whalen': defaultdict(<class 'int'>, {'Jason Trost':
1}), 'Divine': defaultdict(<class 'int'>, {'John Waters': 1}), 'Maggie Cheung': d
efaultdict(<class 'int'>, {'Olivier Assayas': 1}), 'Fereshteh Sadre Orafiay': def
aultdict(<class 'int'>, {'Jafar Panahi': 1}), "Patrick O'Donnell": defaultdict(<c
lass 'int'>, {'Ivan Kavanagh': 1}), 'KÃ´ji Yakusho': defaultdict(<class 'int'>,
{'Kiyoshi Kurosawa': 1}), 'Tatiana Suarez-Pico': defaultdict(<class 'int'>, {'Tad
eo Garcia': 1}), 'Julianna Pitt': defaultdict(<class 'int'>, {'Thomas L. Phillip
s': 1}), 'Shane Carruth': defaultdict(<class 'int'>, {'Shane Carruth': 1}), 'Ian
Gamazon': defaultdict(<class 'int'>, {'Neill Dela Llana': 1}), 'Carlos Gallardo':
defaultdict(<class 'int'>, {'Robert Rodriguez': 1}), 'Richard Jewell': defaultdic
t(<class 'int'>, {'Anthony Vallone': 1}), 'Kerry BishÃ©': defaultdict(<class 'in
t'>, {'Edward Burns': 1}), 'Natalie Zea': defaultdict(<class 'int'>, {'': 1}), 'E
va Boehnke': defaultdict(<class 'int'>, {'Benjamin Roberds': 1}), 'John August':
defaultdict(<class 'int'>, {'Jon Gunn': 1}))
```

**ESTADO:** Completado con éxito

## B.4. Print seleccionado

Demasiada información. Deseamos imprimir únicamente, para cada actor, las colaboraciones que superen un mínimo, dato entrada. Si un actor no tiene ninguna colaboración que supere dicho mínimo, lógicamente no debe mostrarse.

```
In [85]: for actor_name in num_collaborations:
        selected_directors = [
            director_counter
            for director_counter in num_collaborations[actor_name].items()
            if director_counter[1] >= 5 # Cogemos un mínimo de 5 películas, por ej.
        ]
        if selected_directors:
            print(f'{actor_name} -> {selected_directors}')
```

```
Johnny Depp -> [('Gore Verbinski', 5), ('Tim Burton', 6)]
Leonardo DiCaprio -> [('Martin Scorsese', 5)]
Robert De Niro -> [('Martin Scorsese', 7)]
Bill Murray -> [('Wes Anderson', 5)]
Clint Eastwood -> [('Clint Eastwood', 10)]
Woody Allen -> [('Woody Allen', 10)]
```

**ESTADO:** Completado con éxito

## C. Algunos gráficos sencillos [1 punto]

### C.1 Un modelo de gráfica

Vamos a diseñar un modelo de gráfica sencillo que nos sirva para las siguientes representaciones. Tomará como parámetro una lista de pares  $(x, y)$ , y opcionalmente los tres rótulos explicativos que necesitamos incluir. Además, queremos que las etiquetas de las abscisas aparezcan inclinadas, para poder luego mostrar intervalos de edad.

Las pruebas de funcionamiento te darán más información que las explicaciones que pueda yo dar aquí.

```
In [86]: def representar_xxx_yyy(values: list, strings: list = False):
        """
        Representa una gráfica a partir de los valores dados.

        Parameters:
        -----
        values : list
            Una lista de tuplas o listas donde el primer elemento es el valor de
            x y el segundo es el valor de y.
        strings : list, optional
            Una lista de cadenas que se utilizarán para etiquetar los ejes y el
            título de la gráfica. Por defecto es False.

        Returns:
        -----
        None
        """
        x_values = [item[0] for item in values]
        y_values = [item[1] for item in values]

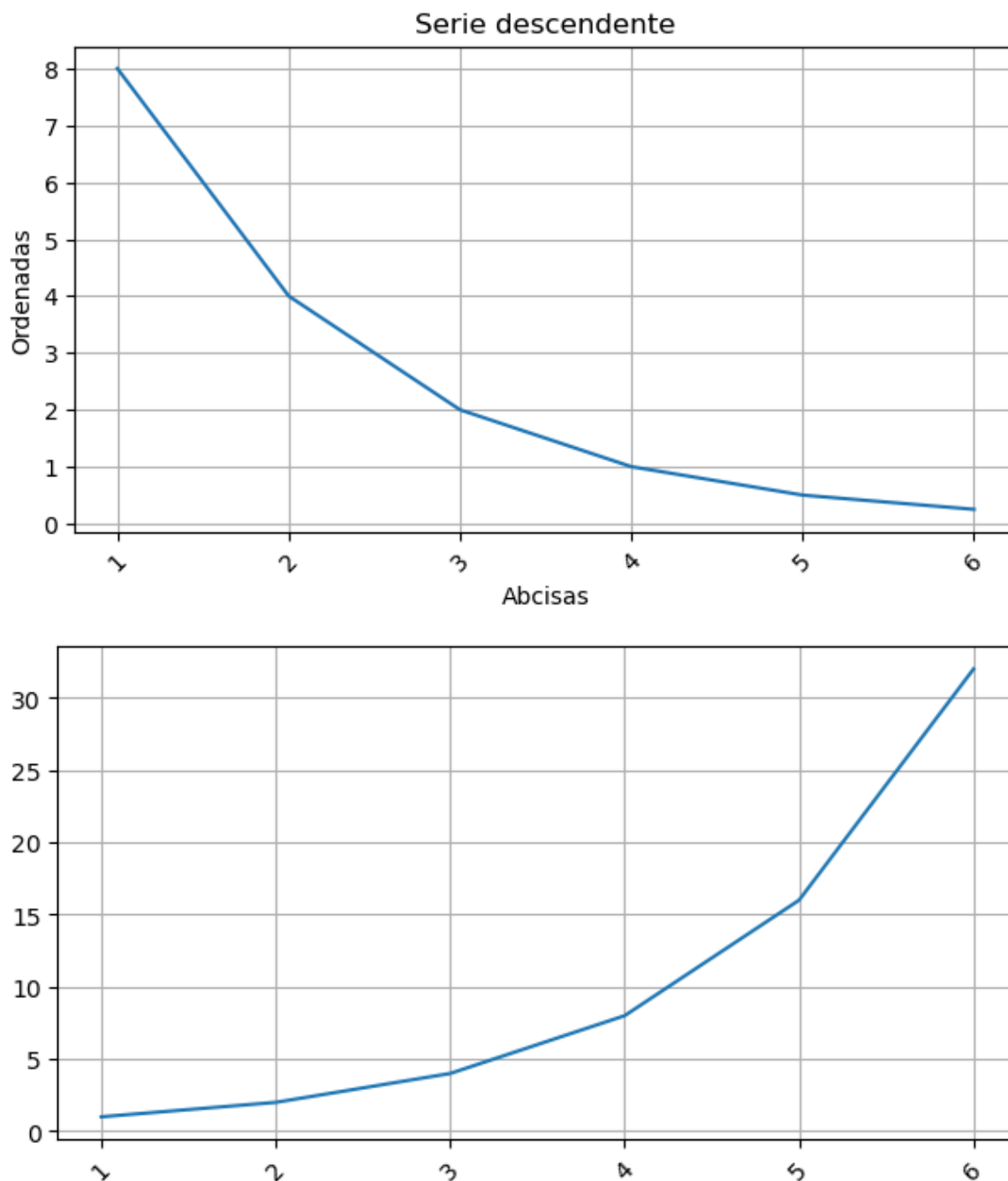
        fig, ax = plt.subplots(figsize=(7.25, 3.75)) # Elegimos un tamaño arbitrario
        ax.plot(x_values, y_values)

        if strings: # Si la entrada de las etiquetas es True, las representamos.
            ax.set(xlabel=strings[2], ylabel=strings[1], title=strings[0])

        plt.xticks(x_values, rotation=45) # Rotamos las etiquetas 45º grados.
        plt.grid() # Añadimos la cuadrícula.
        plt.show()
```

```
In [87]: # Pruebas de funcionamiento:

representar_xxx_yyy([(1, 8), (2, 4), (3, 2), (4, 1), (5, 0.5), (6, 0.25)], ["Ser
representar_xxx_yyy([(1, 1), (2, 2), (3, 4), (4, 8), (5, 16), (6, 32)])
```



**ESTADO:** Completado con éxito

## Una gráfica concreta

Deseamos representar el número de películas de nuestra base de datos que se han producido en un intervalo de años dado.

```
In [88]: def repr_movies_years(dataset: dict, min_year: int, max_year: int):
  """
  Representa la cantidad de películas en un conjunto de datos por año
  dentro de un rango específico.

  Parameters:
  -----
  dataset : dict
      Un diccionario donde las claves son años y los valores son datos
```

```

    relacionados con películas.
    min_year : int
        El año mínimo del rango a considerar.
    max_year : int
        El año máximo del rango a considerar.

    Returns:
    -----
    None
    """

    data = years_num_movies(dataset, min_year, max_year)
    num_movies_sorted = [(year, data[year]) for year in data.keys()]

    return representar_xxx_yyy(num_movies_sorted,
                               ["Number of movies in our dataset",
                                "num. movies in a year",
                                "years"])

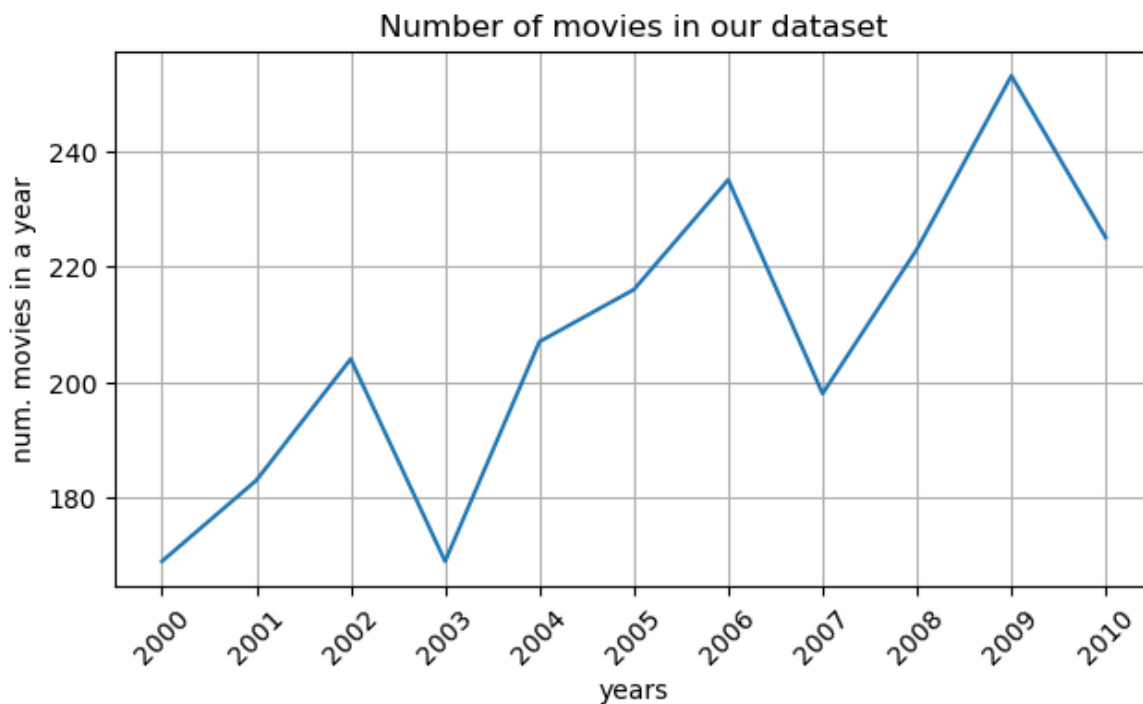
```

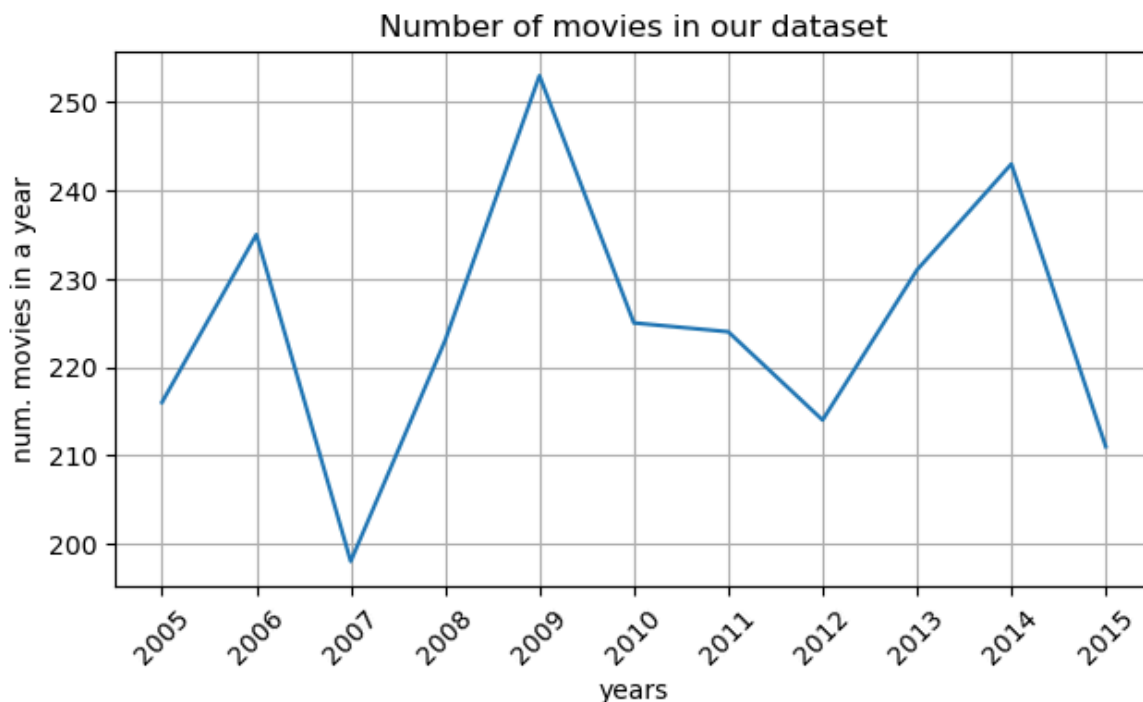
In [89]: *# Test de funcionamiento*

```

repr_movies_years(main_dict_data, 2000, 2010)
repr_movies_years(main_dict_data, 2005, 2015)

```





**ESTADO:** Completado con éxito

## D. Acceso a las urls de imdb y webscraping [2 puntos]

### D.1. Recuperación de las URLs

Con sencillas instrucciones, deseamos recuperar todas las *urls* de las películas de nuestro archivo, mostrando cuántas son, la primera de ellas o las diez primeras por ejemplo:

```
In [90]: # La url se halla en el último elemento de la lista de datos del diccionario.
urls = [item[-1] for item in main_dict_data.values()]

first_url_movie = urls[0]

first_ten_urls = urls[:10]
```

```
In [91]: # Test de funcionamiento

print(len(urls))

print()

print(first_url_movie)

print()

print(first_ten_urls)
```



4919

<http://www.imdb.com/title/tt0499549/>

```
[ 'http://www.imdb.com/title/tt0499549/', 'http://www.imdb.com/title/tt0449088/',
'http://www.imdb.com/title/tt2379713/', 'http://www.imdb.com/title/tt1345836/',
'http://www.imdb.com/title/tt5289954/', 'http://www.imdb.com/title/tt0401729/',
'http://www.imdb.com/title/tt0413300/', 'http://www.imdb.com/title/tt0398286/',
'http://www.imdb.com/title/tt2395427/', 'http://www.imdb.com/title/tt0417741/']
```

**ESTADO:** Completado con éxito

## D.2. Carga de la estructura sintáctica de una URL

Ahora, deseamos extraer el código `html` de una película.

```
In [92]: def soup_movie(url: str):
        """Extraemos el contenido web de una URL dada.

        Parameters:
        -----
        url : str
            La URL de la película sobre la que extraer los datos.

        Returns:
        -----
        BeautifulSoup
            Un objeto BeautifulSoup que contenga los datos extraídos de la web.
        """
        # Es necesario especificar el headers para tener acceso a la web.
        url_content = requests.get(url, headers=HEADERS)
        return BeautifulSoup(url_content.content, 'html.parser')
```

```
In [93]: # Test de funcionamiento
soup = soup_movie(first_url_movie)
print(str(soup)[:1000])
print()
print("... ..")
print()
print(str(soup)[-1000:])
```

```

<!DOCTYPE html>
<html lang="en-US" xmlns:fb="http://www.facebook.com/2008/fbml" xmlns:og="http://
opengraphprotocol.org/schema/"><head><meta charset="utf-8"/><meta content="width=
device-width" name="viewport"/><script>if(typeof uet === 'function'){ uet('bb',
'LoadTitle', {wb: 1}); }</script><script>window.addEventListener('load', (event)
=> {
    if (typeof window.csa !== 'undefined' && typeof window.csa === 'functio
n') {
        var csaLatencyPlugin = window.csa('Content', {
            element: {
                slotId: 'LoadTitle',
                type: 'service-call'
            }
        });
        csaLatencyPlugin('mark', 'clickToBodyBegin', 1729675657633);
    }
})</script><title>Avatar (2009) - IMDb</title><meta content="Avatar: Directed
by James Cameron. With Sam Worthington, Zoe Saldana, Sigourney Weaver, Stephen La
ng. A paraplegic Marine dispatched to the moon Pandora on a unique mission become
s torn between following his orders and pr

... ..
... ..

g(e+c);return!!e}function n(){for(var e=RegExp("^https://(.*\\.(images|ssl-images|
media)-amazon\\.com|"+c.location.hostname+)/images/", "i"),d={},h=0,k=c.performanc
e.getEntriesByType("resource"),l=!1,b,a,m,f=0;f<k.length;f++)if(a=k[f],0<a.transf
erSize&&a.transferSize>a.encodedBodySize&&(b=e.exec(String(a.name)))&&3===b.leng
th){a:{b=a.serverTiming||[];for(a=0;a<b.length;a++)if("provider"===b[a].name){b=b
[a].description;break a}b=void 0}b&&(1||(1=g(b,"_cdn_fr")),
a=d[b]=(d[b]||0)+1,a>h&&(m=b,h=a))}g(m,"_cdn_mp")}d.ue&&"function"===typeof d.ue.
tag&&c.performance&&c.location&&n()),"cdnTagging")(ue_csm>window);

}

/* Δ */
</script>
</div>
<noscript>

</noscript>
<script>window.ue && ue.count && ue.count('CSMLibrarySize', 61463)</script></div>
</body></html>

```

**ESTADO:** Completado con éxito

### D.3. Extracción de algunas piezas de información de una URL

Y ahora, con dicho código, deseamos extraer la siguiente información, referida a la película `first_url_movie`:

- La etiqueta completa del título de la película
- La descripción (sólo el contenido).

- La lista de los actores del *reperto principal*
- La información sobre el presupuesto

```
In [94]: # Obtenemos el título de la película.
title_label = soup.find('title')

# La descripción de la película se halla en la etiqueta <meta>.
description_label = soup.find('meta', {'name': 'description'}).get('content')

# Buscamos todos los actores que se hallen dentro de la 'title-cast'.
actors_label = soup.find('section', {'data-testid': 'title-cast'}).find_all(
    'a', {'class': 'ipc-lockup-overlay ipc-focusable'})

# Extraemos todos los nombres de los actores hallados en la etiqueta anterior.
actors_names = [item.get('aria-label') for item in actors_label]

# Buscamos y extraemos la información respecto al presupuesto de la película.
budget_label = soup.find(
    'li', {'role': 'presentation',
          'class': "ipc-metadata-list__item sc-1bec5ca1-2 eoigIp"})

budget = budget_label.find(
    'span', {'aria-disabled': "false",
            'class': "ipc-metadata-list-item__list-content-item"})
).text
```

```
In [95]: # Test de funcionamiento
print(title_label)
print()
print(description_label)
print()
print(actors_names)
print()
print(budget)
```

<title>Avatar (2009) - IMDb</title>

Avatar: Directed by James Cameron. With Sam Worthington, Zoe Saldana, Sigourney Weaver, Stephen Lang. A paraplegic Marine dispatched to the moon Pandora on a unique mission becomes torn between following his orders and protecting the world he feels is his home.

['Sam Worthington', 'Zoe Saldana', 'Sigourney Weaver', 'Michelle Rodriguez', 'Stephen Lang', 'Giovanni Ribisi', 'Joel David Moore', 'CCH Pounder', 'Wes Studi', 'Laz Alonso', 'Dileep Rao', 'Matt Gerald', 'Sean Anthony Moran', 'Jason Whyte', 'Scott Lawrence', 'Kelly Kilgour', 'James Patrick Pitt', 'Sean Patrick Murphy']

\$237,000,000 (estimated)

**ESTADO:** Completado con éxito

## D.4. Actores que intervienen en una lista de URLs

Necesitamos crear un archivo con los actores del reparto principal de las películas de IMDB, dada la lista de sus URLs.

```
In [96]: def gather_actors(output_file: str, urls: list):
        """Reúne el nombre de los actores partícipes en las películas y los escribe
        en un archivo externo.

        Parameters:
        -----
        output_file : str
            La ruta del archivo sobre la que el documento se escribirá.
        urls : list
            Una lista de URLs sobre las que buscar los actores.
        """
        with open(output_file, 'w', encoding='Latin1') as write_file:
            for url in urls:
                soup = soup_movie(url)
                actors_label = soup.find(
                    'section', {'data-testid': 'title-cast'})
                actors_label.find_all(
                    'a', {'class': 'ipc-lockup-overlay ipc-focusable'})
                # Buscamos el nombre de los actores.
                actors_names = [item.get('aria-label') for item in actors_label]
                for actor in actors_names:
                    write_file.write(f'{actor}\n')
```

```
In [97]: # Test de funcionamiento

        # OJO: esta operación puede llevar bastante tiempo.
        # Para esta prueba, usamos un número limitado de películas.

        gather_actors("actors_3_first_movies.txt", urls[:3])

        ! type actors_3_first_movies.txt
```

Sam Worthington  
Zoe Saldana  
Sigourney Weaver  
Michelle Rodriguez  
Stephen Lang  
Giovanni Ribisi  
Joel David Moore  
CCH Pounder  
Wes Studi  
Laz Alonso  
Dileep Rao  
Matt Gerald  
Sean Anthony Moran  
Jason Whyte  
Scott Lawrence  
Kelly Kilgour  
James Patrick Pitt  
Sean Patrick Murphy  
Johnny Depp  
Orlando Bloom  
Keira Knightley  
Geoffrey Rush  
Jack Davenport  
Bill Nighy  
Jonathan Pryce  
Lee Arenberg  
Mackenzie Crook  
Kevin McNally  
David Bailie  
Stellan Skarsgård  
Tom Hollander  
Naomie Harris  
Martin Klebba  
David Schofield  
Lauren Maher  
Dermot Keaney  
Daniel Craig  
Christoph Waltz  
Léa Seydoux  
Ralph Fiennes  
Monica Bellucci  
Ben Whishaw  
Naomie Harris  
Dave Bautista  
Andrew Scott  
Rory Kinnear  
Jesper Christensen  
Alessandro Cremona  
Stephanie Sigman  
Tenoch Huerta  
Adriana Paz  
Domenico Fortunato  
Marco Zingaro  
Stefano Elfi DiClaudia

**ESTADO:** Completado con éxito

In [ ]: *# La siguiente llamada llevaría un tiempo realmente largo:*

```

reloj_inicio = time.time()
gather_actors("actors_all_movies.txt", urls)
reloj_fin = time.time()

print("Tiempo invertido: %s segundos." % (reloj_fin - reloj_inicio))

```

## E. Pandas [2 puntos]

### E.1. El primer paso es la carga del archivo en un dataframe

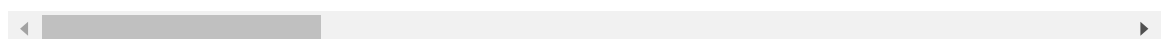
In [98]: *# Aquellas funciones que sean triviales las escribiremos con Lambda.*  
 load\_dataframe = **lambda** file\_name: pd.read\_csv(file\_name)

In [99]: *# Test de funcionamiento*  
 tabla\_completa = load\_dataframe(MOVIES\_DATA)  
 tabla\_completa

Out[99]:

	color	director_name	num_critic_for_reviews	duration	director_facebook_likes	ac
0	Color	James Cameron	723.0	178.0	0.0	
1	Color	Gore Verbinski	302.0	169.0	563.0	
2	Color	Sam Mendes	602.0	148.0	0.0	
3	Color	Christopher Nolan	813.0	164.0	22000.0	
4	NaN	Doug Walker	NaN	NaN	131.0	
...	...	...	...	...	...	
5038	Color	Scott Smith	1.0	87.0	2.0	
5039	Color	NaN	43.0	43.0	NaN	
5040	Color	Benjamin Roberds	13.0	76.0	0.0	
5041	Color	Daniel Hsia	14.0	100.0	0.0	
5042	Color	Jon Gunn	43.0	90.0	16.0	

5043 rows × 28 columns



**ESTADO:** Completado con éxito

### E.2. Tabla de los campos principales

A partir de la tabla anterior, construimos otra con sólo algunos de los campos:

```
In [100... # FIELDS_SELECTED ha sido mencionado al principio en la celda de Las constantes.
fields_selected_dataframe = lambda dataframe: dataframe[FIELDS_SELECTED]
```

```
In [101... # test de comprobación

tabla_breve = fields_selected_dataframe(tabla_completa)

tabla_breve
```

Out[101...

	movie_title	color	director_name	language	country	actor_1_name	
0	Avatar	Color	James Cameron	English	USA	CCH Pounder	http://www.ir
1	Pirates of the Caribbean: At World's End	Color	Gore Verbinski	English	USA	Johnny Depp	http://www.ir
2	Spectre	Color	Sam Mendes	English	UK	Christoph Waltz	http://www.ir
3	The Dark Knight Rises	Color	Christopher Nolan	English	USA	Tom Hardy	http://www.ir
4	Star Wars: Episode VII - The Force Awakens ...	NaN	Doug Walker	NaN	NaN	Doug Walker	http://www.ir
...	...	...	...	...	...	...	...
5038	Signed Sealed Delivered	Color	Scott Smith	English	Canada	Eric Mabius	http://www.ir
5039	The Following	Color	NaN	English	USA	Natalie Zea	http://www.ir
5040	A Plague So Pleasant	Color	Benjamin Roberds	English	USA	Eva Boehnke	http://www.ir
5041	Shanghai Calling	Color	Daniel Hsia	English	USA	Alan Ruck	http://www.ir
5042	My Date with Drew	Color	Jon Gunn	English	USA	John August	http://www.ir

5043 rows × 7 columns

ESTADO: Completado con éxito

E.3. Columnas de una tabla

¿Cuáles son las columnas de nuestra `tabla_breve` ?

```
In [102... tabla_breve.columns
```

```
Out[102... Index(['movie_title', 'color', 'director_name', 'language', 'country',  
        'actor_1_name', 'movie_imdb_link'],  
      dtype='object')
```

**ESTADO:** Completado con éxito

## E.4. Campos missing

Algunos campos muestran un valor `NaN`. Deseamos cambiarlo por una cadena de caracteres: `"Desc"`.

```
In [103... # Se producirá un warning. En el test de comprobación se da igualmente lo pedido  
tabla_breve.fillna('Desc', inplace = True)
```

C:\Users\Usuario\AppData\Local\Temp\ipykernel\_9028\3951709524.py:2: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
tabla_breve.fillna('Desc', inplace = True)
```

```
In [104... # Test de comprobación  
  
tabla_breve
```



Out[104...

	movie_title	color	director_name	language	country	actor_1_name	
0	Avatar	Color	James Cameron	English	USA	CCH Pounder	http://www.ir
1	Pirates of the Caribbean: At World's End	Color	Gore Verbinski	English	USA	Johnny Depp	http://www.ir
2	Spectre	Color	Sam Mendes	English	UK	Christoph Waltz	http://www.ir
3	The Dark Knight Rises	Color	Christopher Nolan	English	USA	Tom Hardy	http://www.ir
4	Star Wars: Episode VII - The Force Awakens ...	Desc	Doug Walker	Desc	Desc	Doug Walker	http://www.ir
...	...	...	...	...	...	...	...
5038	Signed Sealed Delivered	Color	Scott Smith	English	Canada	Eric Mabius	http://www.ir
5039	The Following	Color	Desc	English	USA	Natalie Zea	http://www.ir
5040	A Plague So Pleasant	Color	Benjamin Roberds	English	USA	Eva Boehnke	http://www.ir
5041	Shanghai Calling	Color	Daniel Hsia	English	USA	Alan Ruck	http://www.ir
5042	My Date with Drew	Color	Jon Gunn	English	USA	John August	http://www.ir

5043 rows × 7 columns

**ESTADO:** Completado con éxito

## E.5. Director $\rightarrow$ películas y número de películas

Función que averigua la lista de títulos de películas de un director dado:

In [105...

```
def titulos_de_director_df(table: pd.DataFrame, director_name: str):
    """Devuelve las películas dirigidas por un director específico en el
    DataFrame.

    Parameters:
    -----
    table : pd.DataFrame
        El DataFrame conteniendo los datos, donde el elemento 0 es el nombre
```

```

        de la película.
        director_name : str
            El nombre del director del que extraer los datos.

        Returns:
        -----
        DataFrame
            Un DataFrame que solo contenga aquellas películas dirigidas por el
            director especificado.
        """
        return table.loc[table['director_name'] == director_name].iloc[:, [0]]

```

```

In [106... # Test de comprobación:

tabla_tits = titulos_de_director_df(tabla_breve, "James Cameron")
tabla_tits

```

```

Out[106...

```

	movie_title
0	Avatar
26	Titanic
288	Terminator 2: Judgment Day
291	True Lies
606	The Abyss
2486	Aliens
3575	The Terminator

**ESTADO:** Completado con éxito

```

In [107... list_tits = tabla_tits["movie_title"].to_list()
print(list_tits)

['Avatar\xa0', 'Titanic\xa0', 'Terminator 2: Judgment Day\xa0', 'True Lies\xa0',
'The Abyss\xa0', 'Aliens\xa0', 'The Terminator\xa0']

```

**ESTADO:** Completado con éxito

Deseamos saber qué directores han dirigido el máximo número de películas, junto con ese número de películas.

```

In [108... def directors_max_movies_df(table: pd.DataFrame):
    """Devuelve el director con el número máximo de películas dirigidas de un
    DataFrame.

    Parameters:
    -----
    table : pd.DataFrame
        El DataFrame conteniendo información sobre las películas.

    Returns:
    -----

```

```

tuple
    Una tupla que contenga el nombre del director que ha dirigido más
    películas, junto a dicho número.
"""
# Filtramos las entradas 'Desc' en 'director_name' introducidas antes.
filtered_table = table[table['director_name'] != 'Desc']

# Obtenemos la frecuencia.
summary = filtered_table['director_name'].value_counts()

max_value = max(summary)

for name, value in zip(summary.index, summary):
    if value == max_value:
        return ([name], value)

```

```

In [109... # test de comprobación:

print(directors_max_movies_df(tabla_breve))

(['Steven Spielberg'], 26)

```

**ESTADO:** Completado con éxito

## Parte F. Un cálculo masivo con map-reduce [0,5 puntos]

En este apartado se ha de realizar un programa aparte, *basado en la técnica de map-reduce*, que calcule, para cada idioma, en qué países en que se han producido películas y la suma de los presupuestos de dichas películas. Cuando el idioma o el país o el presupuesto no se conozcan, no se considerará esta película.

```

C:\...> python language_budget_countries.py -q
algunos_campos.txt

```

El programa funcionará necesariamente con la técnica map-reduce, que podemos poner en juego con la librería `mrjob`.

El funcionamiento del mismo se puede activar también desde aquí:

NOTA DEL ESTUDIANTE: he decidido escribir el archivo usando magic lines por su comodidad con `%%writefile`.

```

In [54]: %%writefile language_budget_countries.py

# Deben importarse las librerías necesarias dentro de la propia script.
from mrjob.job import MRJob
from mrjob.step import MRStep

class MRLanguageBudgetCountries(MRJob):
    """MapReduce para agrupar el presupuesto total de películas según el
    idioma. También, se incluirá una lista de aquellas películas que

```

comparten el mismo idioma. El proceso se lleva a cabo mediante un mapper que procesa los datos y un reducer que agrega los resultados en la output final.

Attributes:

-----

None

"""

```
def mapper(self, _, line: str):
    """Procesa cada línea del archivo que lee para devolver el idioma,
    el país y el presupuesto de la película.
```

Primero, divide la línea con separador '|' en campos aislados. De ahí se extraen los valores del idioma y país. Se filtran los datos en función de si se encuentran vacíos o su formato no es el deseado. Finalmente, se devuelven el idioma, país y presupuesto de la película.

Parameters:

-----

\_ : None

No usada.

line : str

La línea que se lee como entrada a la función.

Yields:

-----

tuple

Una tupla donde el primer elemento es el idioma de la película, y otra tupla que contenga el país y presupuesto de esta.

"""

*# Se divide la línea en campos individuales.*

fields = line.split('|')

*# No se devuelve nada si faltan campos.*

if len(fields) < 5:

return

language = fields[2].strip()

country = fields[3].strip()

if language == '' or country == '':

return

*# Se ignoran líneas donde el presupuesto no sea un número.*

try:

budget = float(fields[-1].strip())

except ValueError:

return

*# Se utiliza yield para usar estos datos en el reducer.*

yield language, (country, budget)

```
def reducer(self, language: str, values: tuple):
```

"""Suma todos los presupuestos de las películas con el mismo idioma, devolviendo el idioma, una lista de los países donde se han grabado y su presupuesto total sumado.

Parameters:

```

-----
language : str
    El idioma en el que se ha grabado la película.
values : tuple
    Iterador de tuplas, donde cada una contendrá el país y
    el presupuesto de la película (directamente de la output
    del mapper).

Yields:
-----
tuple
    Tupla donde el primer elemento es el idioma y el segundo
    una lista cuyo primer elemento es una lista de los países
    con dicho idioma y el segundo elemento la suma total de
    presupuestos.
"""
countries = set() # Usamos un set para que no se repitan países.
total_budget = 0

for country, budget in values:
    countries.add(country) # Añadimos el país al set.
    total_budget += budget # Sumamos el presupuesto al total.

yield language, [list(countries), total_budget]

def steps(self):
    """Define el orden de las tareas a ejecutarse en el MapReduce.
    Devuelve una lista de instancias de MRJob.

    Returns:
    -----
    list
        Una lista conteniendo las tareas a ejecutarse en el MapReduce.
    """
    return [MRStep(mapper=self.mapper, reducer=self.reducer)]

if __name__ == '__main__':
    MRLanguageBudgetCountries.run() # Se ejecuta el programa.

```

Overwriting language\_budget\_countries.py

```
In [ ]: # Hagamos una llamada al programa de consola desde aquí:
```

```
! python language_budget_countries.py -q algunos_campos.txt
```

```
In [ ]: # Para que el resultado se almacene en un archivo:
```

```
! python language_budget_countries.py -q algunos_campos.txt > language_countries
```

**ESTADO:** Completado con éxito

La siguiente celda me permite ver tu programa cómodamente desde aquí.

```
In [ ]: def print_file(filename):
        with open(filename, "r") as f:
            for line in f:
                print(line, end="")
```

```
print_file("language_budget_countries.py")
```

## Parte G. Un apartado libre [0.5 puntos]

Realiza un análisis sobre el presupuesto de aquellas películas cuya entrada en la celda *budget* del dataframe no sea nula. Agrupa las películas en rangos de presupuesto y representa el conteo de ellas en un gráfico de barras. También, escribe el género predominante en cada grupo.

```
In [50]: def barplot(
df: pd.DataFrame, bins: int, lower_percent: float, upper_percent: float
):
    """
    Se crea un diagrama de barras para visualizar el número de películas
    dentro de un cierto rango de presupuesto y el género predominante
    de las películas dentro de dicho rango. Dado que el espectro de
    presupuestos de las películas es muy variado, tan solo nos
    centramos en aquel rango comprendido entre lower_percent
    y upper_percent.

    Parameters:
    -----
    df : pd.DataFrame
        DataFrame que contenga información sobre las películas (presupuesto
        y género).
    bins : int
        Número de barras a representar en el diagrama de barras.
    lower_percent : float
        Límite inferior sobre el que analizar los datos.
    upper_percent : float
        Límite superior sobre el que analizar los datos.
    """
    # Filtramos aquellos películas cuyo presupuesto no sea nulo.
    data = df[df['budget'].notna()]

    # Calculamos los percentiles, limitando el rango de presupuestos a analizar.
    quantile_lower = data['budget'].quantile(lower_percent)
    quantile_upper = data['budget'].quantile(upper_percent)

    filtered_data = data[
        (data['budget'] <= quantile_upper) & (data['budget'] >= quantile_lower)
    ]

    step = (quantile_upper - quantile_lower) / bins

    # Se crean las etiquetas para agrupar las películas según el presupuesto.
    labels = [
        f'{value * 10**-6:.0f} - {(value + step) * 10**-6:.0f}'
        for value in np.linspace(quantile_lower, quantile_upper - step, bins)
    ] # Se utilizan las unidades de millones de $ (*10**-6).

    # Se agrupan los presupuestos según unos rangos en la columna 'budget_range'
    filtered_data['budget_range'] = pd.cut(
        filtered_data['budget'],
        bins,
        labels=labels)
```

```

# Se calcula la frecuencia de cada rango de presupuestos.
bin_counts = filtered_data['budget_range'].value_counts().sort_index()

# Se extrae el primer género y se almacena en la columna 'genre'.
filtered_data['genre'] = filtered_data['genres'] \
    .apply(lambda x: x.split('|')[0])

def predominant_genre(group: pd.DataFrame):
    """Devuelve la moda de la columna 'genre' de un DataFrame.

    Parameters:
    -----
    group : pd.DataFrame
        DataFrame sobre el que se calculará la moda.

    Returns:
    -----
    mode_value : str
        El valor de la moda. En el presente caso, ya que la columna
        'genre' está compuesta de strings, la output será un string.

    """
    mode_value = group['genre'].mode()[0]
    return mode_value

predominant_genres = filtered_data.groupby('budget_range') \
    .apply(predominant_genre)

# Se crean las etiquetas que muestren el rango de presupuesto y el género.
labels_plot = [
    f'{labels[i]} {predominant_genres[i]}' for i in range(len(labels))
]

fig, ax = plt.subplots(figsize=(10, 5))
ax.bar(bin_counts.index, bin_counts.values)
ax.set(title='Número de películas según su presupuesto',
       xlabel='Presupuesto (millones de $)',
       ylabel='Número de películas')

# Se establecen las etiquetas customizadas antes.
plt.xticks(ticks=np.arange(len(labels)), labels=labels_plot, rotation=90)

# Se crea una cuadrícula en el eje vertical.
plt.grid(axis='y', linestyle='--', alpha=0.7)

plt.show()

```

```

In [51]: # test de comprobación:
# BINS, LOWER_PERCENT y UPPER_PERCENT se han definido en la celda de constantes.
barplot(tabla_completa, BINS, LOWER_PERCENT, UPPER_PERCENT) # Emite un warning.

```

```
C:\Users\Usuario\AppData\Local\Temp\ipykernel_9028\3742435354.py:44: SettingWithCopyWarning:
```

A value is trying to be set on a copy of a slice from a DataFrame.

Try using `.loc[row_indexer,col_indexer] = value` instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
filtered_data['budget_range'] = pd.cut(
```

```
C:\Users\Usuario\AppData\Local\Temp\ipykernel_9028\3742435354.py:53: SettingWithCopyWarning:
```

A value is trying to be set on a copy of a slice from a DataFrame.

Try using `.loc[row_indexer,col_indexer] = value` instead

See the caveats in the documentation: [https://pandas.pydata.org/pandas-docs/stable/user\\_guide/indexing.html#returning-a-view-versus-a-copy](https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
filtered_data['genre'] = filtered_data['genres'] \
```

```
C:\Users\Usuario\AppData\Local\Temp\ipykernel_9028\3742435354.py:74: FutureWarning: The default of observed=False is deprecated and will be changed to True in a future version of pandas. Pass observed=False to retain current behavior or observed=True to adopt the future default and silence this warning.
```

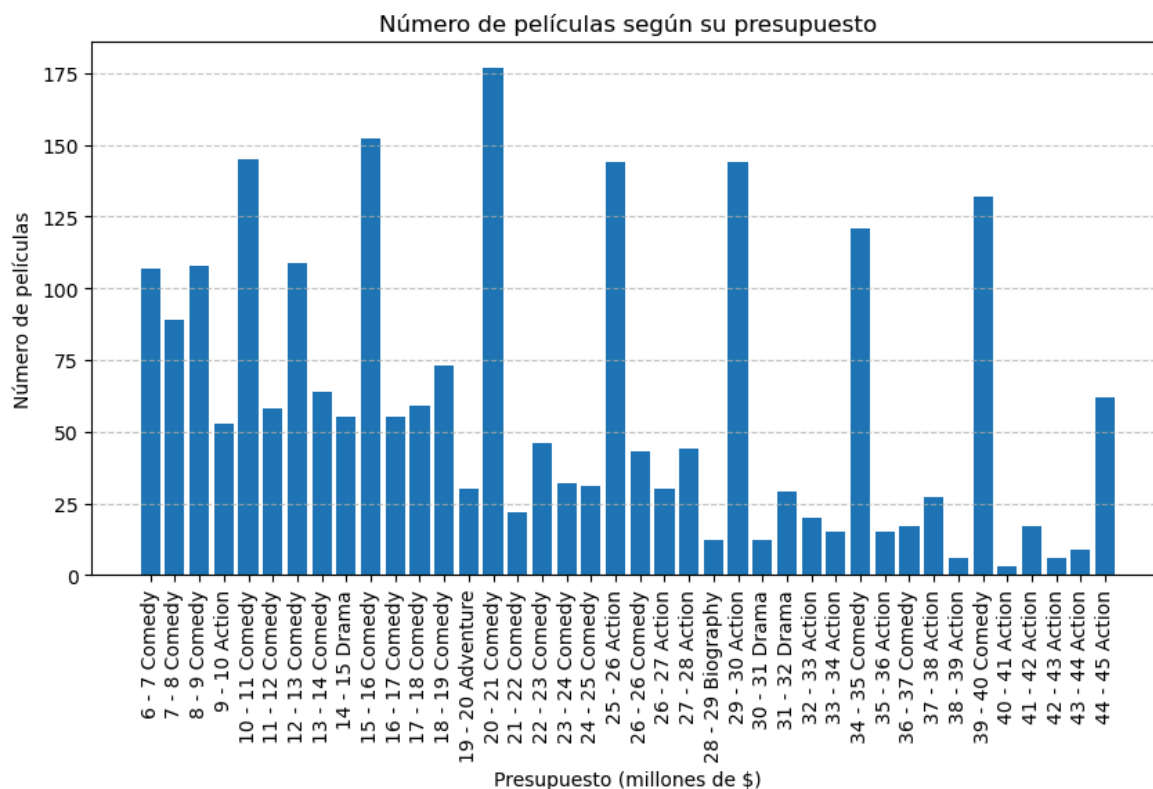
```
predominant_genres = filtered_data.groupby('budget_range') \
```

```
C:\Users\Usuario\AppData\Local\Temp\ipykernel_9028\3742435354.py:75: DeprecationWarning: DataFrameGroupBy.apply operated on the grouping columns. This behavior is deprecated, and in a future version of pandas the grouping columns will be excluded from the operation. Either pass `include_groups=False` to exclude the grouping columns or explicitly select the grouping columns after groupby to silence this warning.
```

```
.apply(predominant_genre)
```

```
C:\Users\Usuario\AppData\Local\Temp\ipykernel_9028\3742435354.py:79: FutureWarning: Series.__getitem__ treating keys as positions is deprecated. In a future version, integer keys will always be treated as labels (consistent with DataFrame behavior). To access a value by position, use `ser.iloc[pos]`
```

```
f'{labels[i]} {predominant_genres[i]}' for i in range(len(labels))
```





**ESTADO:** Completado con éxito

Cabe destacar que aquellas películas con un mayor presupuesto son aquellas dedicadas a la **Acción**, mientras que aquellas películas de bajo presupuesto son mayoritariamente del género **Comedia** .

## Datos personales

- **Apellidos:** Mestre Fons
- **Nombre:** Bartolomé
- **Email:** tolo201311@gmail.com (bmestre@ucm.es)
- **Fecha:** Lunes, 21 de Octubre de 2024

## Ficha de autoevaluación

A continuación se procede a realizar una autoevaluación de las soluciones propuestas por el estudiante en una tabla:

Apartado	Calificación	Comentario
a)	2.5 / 2.5	Completamente resuelto
b)	2.0 / 2.0	Completamente resuelto
c)	1.5 / 1.5	Completamente resuelto
d)	1.0 / 1.0	Completamente resuelto
e)	2.0 / 2.0	Completamente resuelto
f)	0.5 / 0.5	Completamente resuelto
g)	0.3 / 0.5	Completamente resuelto
<b>Total</b>	<b>9.9 / 10.0</b>	<b>Excelente</b>

El estudiante ha conseguido escribir satisfactoriamente los códigos necesarios para ejecutar las tareas pedidas. Se ha logrado optimizar el código con el fin de mejorar su legibilidad, siguiendo las directrices marcadas por el *pep 8* a la hora de escribir el código. En mi opinión, el programa diseñado en la sección *Parte G. Un apartado libre [0.5 puntos]* podría haberse mejorado para lograr una mayor legibilidad. En resumen, estoy muy contento por el trabajo que he realizado en esta entrega al haber podido poner en práctica los conocimientos adquiridos en las secciones teóricas.

## Ayuda recibida y fuentes utilizadas

El estudiante se ha servido de las siguientes fuentes de información a la hora de realizar el presente trabajo:

- Apuntes personales realizados con los ejercicios del temario.
- Documentación de la libre *mrjob* : [link](#)
- Documentación sobre la escritura en HTML : [link](#)

In [ ]: