UNIVERSITÀ DI PISA

DATA SCIENCE & BUSINESS INFORMATICS

DIPARTIMENTO DI INFORMATICA

Laboratory of Data Science

Progetto Gruppo 14

Studenti

Carlo PALADINO 537650

Francesco SALERNO 534622

Alessandro BONINI 604482

Part 1

Assignment 0

Lo schema del database è stato sviluppato su SQL Server Management Studio. Per quanto riguarda i tipi dei diversi attributi sono state effettuate le seguenti scelte:

- per gli attributi del tipo stringa è stato utilizzato char(100)
- per gli attributi del tipo numerico è stato utilizzato int o float a seconda dell' evenienza.

Sono state definite le chiavi primarie e esterne come suggerito dallo schema fornito, in particolare la connessione tra la tabella *Player* e *Match* è avvenuta con due relazioni differenti: tra la chiave primaria *id_player* e quella esterna *winner_id* e ancora, tra *id_playe*r e loser_id.

Assignment 1

Questo assignment si può suddividere in due fasi svolte in sequenza, nella prima, partendo dai quattro file csv (*tennis.csv, male_players.csv, female_players.csv and country.csv*) abbiamo preparato cinque nuovi file contenenti i dati organizzati in modo da poter riempire le tabelle create in precedenza nel database.

- *Tournament.csv*: sono stati selezionati gli attributi di interesse da tennis.csv
- *Date.csv*: utilizzando l'attributo *tourney_date* da tennis.csv e creando gli attributi *year, month, day e quarter.*
- *Match.csv*: è stato creato l'attributo *match_id* con l'utilizzo di *tourney_id* e *match_num* e sono stati presi gli altri attributi di interesse da tennis.csv
- *Geography.csv*: creato con l'utilizzo del file csv messo a disposizione su didawiki per la lingua e il file geography.csv. Sono stati selezionati *country_ioc, country, continen*t e *language* come attributi ed è stata fatta una leggera pulizia dei dati.
- *Player.csv*: file creato con l'utilizzo di male_players.csv e female_players.csv per ricavare l'attributo *sex* e sono selezionati gli attributi di interesse avvalendosi di un dizionario, inoltre anche qui è stata fatta una prima pulizia dei dati.

La seconda parte è stata dedicata invece alla pulizia dei dati, per questo scopo ci siamo avvalsi della libreria pandas. Per ogni file sopra descritto è stata creata una sua versione *_cleaned* in cui sono stati gestiti i missing values, i duplicati, gli outliers e le varie incongruenze. Alla fine di questo processo abbiamo ottenuto cinque file csv corrispondenti alle tabelle create con SQL Server Management Studio.

Assignment 2

Per popolare il database è stato scritto un programma per ogni tabella (uploadCSV_<nometabella>.py), ognuno dei quali ha la medesima struttura:

• lettura del file csv, utilizzando il metodo read_csv;

• apertura della connessione al database “'Group_14_DB” e creazione del cursore;

• iterazione sulle righe del file;

• scrittura della query in forma parametrica;

• chiusura file, cursore e connessione

Parte 2

Assignment 0

For every tournament, the players ordered by number of matches won.

Il primo nodo inserito nell’attività flusso di dati è un origine OLE DB necessario per accedere alla fact table Match. Successivamente è stato effettuato un group by su winner_id e tourney_id ed un count su match_id, rinominato *vittorie*. Quindi, è stato ordinato il risultato per tourney_id con tipo di ordinamento crescente e vittorie, con tipo di ordinamento decrescente e alias di output *vittorie_giocatore*. Il risultato ottenuto è stato salvato su un file in formato .txt.

Di seguito vengono mostrate le prime righe:

```
tourney_id,winner_id,vittorie_giocatore
2016-0083,200282,7
2016-0083,104327,5
2016-0083,105906,5
2016-0083,105047,3
2016-0083,106072,3
2016-0083,106214,3
```

Assignment 1

A tournament is said to be ”worldwide” if no more than 30% of the participants come from the same continent. List all the worldwide tournaments.

Per questo assignment, gli steps svolti sono stati i seguenti:

- Collezionare tutti i player_id combinando mediante unione winners e losers
- Calcolare il numero totale di players per tournament e continent
- Calcolare il numero totale di players per tournament
- Calcolare tutti i tornei dove la percentuale massima di players provenienti dallo stesso continente è <30%

Il processo ideato su SSIS è composto, quindi, da quattro parti principali:

- La prima parte è necessaria per collezionare tutti i player_id. E' stato inserito un nodo origine OLE DB per accedere alla fact table Match, successivamente un multicast con doppio ordinamento su tourney_id (con tipo di ordinamento crescente) e infine una unione con input 1 winner_id e input 2 loser_id.
- Per risolvere il secondo punto, viene eseguita una ricerca con la tabella Player per ottenere la country di ogni player e in seguito un'altra ricerca per ottenere il continente da associare ad ogni country. Successivamente viene eseguita l'aggregazione per tourney_id e continent e il count su player_id per ottenere il numero totale di players per continente. A questo punto, avendo tutte le informazioni che ci servono, eseguiamo un multicast per svolgere operazioni differenti.
- Per calcolare il numero totale di players per tournament eseguiamo, grazie all' operazione aggregazione, un group by su tourney_id e una somma su TotalPlayerContinent per ottenere *TotalPlayerByTournament* .Arrivati a questo punto viene eseguito un ordinamento di tipo crescente su tourney_id per i due flussi paralleli ottenuti dal multicast e in seguito viene eseguito un merge join.
- Successivamente, convertiamo *TotalPlayerByContinent* e TotalPlayerByTournament come valori a virgola mobile e li dividiamo per ottenere il valore *Average* (TotalPlayerContinent / TotalPlayerByTournament). Infine, tramite l'operatore aggregazione eseguiamo una group by per tourney_id e l'operazione Massimo su Average. Questo perchè, nella suddivisione condizionale, andremo a specificare la condizione "per ogni torneo, se il rapporto più alto tra il numero di players dello stesso continente e il numero totale di players, è minore del 30%, allora aggiungi tale torneo al file finale". Il risultato ottenuto è stato salvato su un file in formato .txt.

Di seguito vengono mostrate le righe ottenute::

```
tourney_id,Average
2019-W-ITF-RSA-01A-2019,0.27419356
2018-W-FC-2018-G2-AO-A-M-NZL-LBN-01,0.25
```

Assignment 2

For each country, list all the players that won more matches than the average number of won matches for all players of the same country.

Per questo assignment, gli steps svolti sono stati i seguenti:

- Calcolare le vittorie totali per giocatore
- Calcolare le vittorie totali per ogni country
- Calcolare i giocatori totali per ogni country
- Selezionare tutti quei giocatori per cui le vittorie totali sono maggiori del rapporto tra totale vittorie per country e giocatori totali per tale country.

Il primo nodo inserito nell'attività flusso di dati è un origine OLE DB necessario per accedere alla fact table Match. Successivamente viene eseguita una ricerca con la tabella Player per

ottenere la colonna country. Successivamente viene applicato un multicast per creare due flussi paralleli. Il flusso di sinistra prosegue con un' aggregazione. Vengono eseguite due group by su winner_id e country e un count su match_id, denominato *PlayerWins*.

Il flusso destro invece, prosegue con un' aggregazione in cui viene eseguito un group by su country ed un count su match_id denominato *Vittorie_Per_country.* Successivamente, entrambe i flussi vengono ordinati su country in modo crescente per un successivo merge. Prima, però, il flusso destro esegue un' operazione di merge un'altra volta con la tabella Player. Tale tabella, viene caricata tramite origine OLE DB, viene eseguita, tramite aggregazione, un group by su country ed un count su id_player per determinare i giocatori totali per ogni country. Infine viene eseguito un ordinamento su country di tipo crescente. A questo punto, viene eseguita una merge join e vengono selezionate come colonne *Country, Vittorie_Per_country* e *PlayerPerCountry*. Poi, viene creata la colonna derivata *AverageWins*, ottenuta come rapporto tra Vittorie_Per_country e PlayerPerCountry. Successivamente, viene eseguita un merge join fra tale flusso e il flusso sinistro uscente dal primo multicast. Infine tramite una suddivisione condizionale, vengono salvati su un file .txt, tutti quei giocatori per cui PlayerWins > AverageWins.

Di seguito vengono mostrate le prime righe:

```
winner_id,PlayerWins,AverageWins,country
213972,78,10,ALG
214604,93,19,ARG
106057,68,19,ARG
144821,28,19,ARG
104216,55,19,ARG
206345,113,19,ARG
```

Parte 3

Assignment 0

Build a datacube from the data of the tables in your database, defining the appropriate hierarchies for time and geography. Use the rank and rank points of the winner and loser as measure.

In questa prima fase è stato costruito il cubo olap Group 14 DB_Parte3, che presenta due dimensioni: *Tourney* e *Player*. All'interno di Tourney, troviamo la gerarchia non flat *YearQuarterMonthTourneyId* composta da Year -> Quarter -> Month -> Tourney_id.

All'interno di Player invece, troviamo la gerarchia non flat *ContinetCountryIDPlayer* composta da Continent -> Country -> IDPlayer.

Assignment 1

Show the player that lost the most matches for each country

```
with member rnk as
rank(([Loser].[Country].currentmember,[Loser].[ContinetCountryIDPlayer].currentmember),
([Loser].[Country].currentmember,[Loser].[ContinetCountryIDPlayer].[Id Player] ),
[Measures].[n_match])

select [Measures].[n_match] on columns,
nonempty(filter(([Loser].[Country].[Country],[Loser].[ContinetCountryIDPlayer].[Id Player]), rnk = 1)) on rows
from [Group 14 DB_Parte3]
```

Per risolvere il seguente assignment, è stata creata una nuova misura *rnk ,* che rappresenta l'ordinamento tramite funzione *rank*, del numero di partite perse da ogni giocatore (perdente) per ogni country. Sulle colonne è stata inserita la misura *n_match* che sarebbe il conteggio delle partite, mentre sulle righe, per ogni riga, è presente la *country*, l'*Id_player* e il *numero di partite del giocatore* che, nell'ordinamento, risulta con rank uguale a 1,cioè quello che ha perso il numero maggiore di partite.

Di seguito vengono mostrate le prime righe:

		n_match
Algeria	Ines Ibbou	63
Andorra	Victoria Jimenez Kasintseva	19
Argentina	Renzo Olivo	145
Armenia	Ani Amiraghyan	31
Australia	Marc Polmans	123
Austria	Sebastian Ofner	125
Bahamas	Kerrie Cartwright	12
Barbados	Darian King	90
Belarus	Uladzimir Ignatik	131
Belgium	Kimmer Coppejans	137

Assignment 2

For each tournament, show the loser with the lowest total loser rank points

```
SELECT [Measures].[Loser Rank Points] ON COLUMNS,
GENERATE(([Tourney].[Tourney Id].[Tourney Id], [Tourney].[Year].[Year]),
BOTTOMCOUNT(([Tourney].[Tourney Id].CURRENTMEMBER, [Tourney].[Year].CURRENTMEMBER,
nonempty([Loser].[Id Player].[Id Player])),
1,
[Measures].[Loser Rank Points])) ON ROWS
FROM [Group 14 DB_Parte3]
```

Per risolvere il seguente assignment, sulle colonne è stata inserita la misura *Loser Rank Points*, mentre sulle righe, tramite funzione *Generate*, per ogni torneo, per ogni anno, viene eseguito un *BottomCount*, cioè viene scelto il loser con misura Loser Rank Points più bassa.

Di seguito vengono mostrate le prime righe:

			Loser Rank Points
Abu Dhabi	2021	Makoto Ninomiya	20
Acapulco	2016	Renata Zarazua	45
Acapulco	2017	Giuliana Olmos	47
Acapulco	2018	Alan Fernando Rubio Fierros	1
Acapulco	2019	Luis Patino	6
Acapulco	2020	Lucas Gomez	10
Acapulco	2021	Luis Patino	15
Adelaide	2020	Mikalai Haliak	10
Adelaide	2021	Kimberly Birrell	41
Agadir $15K	2017	Linda Puppendahl	3

Assignment 3

For each tournament, show the loser with the highest ratio between his loser rank points and the average winner rank points of that tournament.

```
with member AVG_Winner_rank_points as
(([Tourney].[Tourney Id].CURRENTMEMBER, [Loser].[Id Player].[All]),  [Measures].[Winner Rank Points]) /
(([Tourney].[Tourney Id].CURRENTMEMBER, [Loser].[Id Player].[All]), [Measures].[n_match] )

member Loser_Rank_Points as
([Tourney].[Tourney Id].CURRENTMEMBER, [Measures].[Loser Rank Points])

member Ratio as
Loser_Rank_Points / AVG_Winner_rank_points

select { AVG_Winner_rank_points, Loser_Rank_Points, Ratio}  on columns,
GENERATE([Tourney].[Tourney Id].[Tourney Id],
TOPCOUNT(([Tourney].[Tourney Id].CURRENTMEMBER,[Tourney].[Year].[Year], [Loser].[Id Player].[Id Player]), 1, Ratio )) on rows
from [Group 14 DB_Parte3]
```

Per risolvere il seguente assignment, sono state create delle nuove misure: *AVG_winner_rank_points, Loser_Rank_Points* e *Ratio. AVG_winner_rank_points*, definisce per ogni Loser di ogni torneo il rapporto tra *Winner_rank_point*s e il *conteggio di Match* che ha giocato. *Loser_Rank_Points* invece, definisce il totale dei Loser Rank Points di ogni giocatore. Infine, *Ratio* viene calcolato come Loser_Rank_Points / AVG_winner_rank_points.

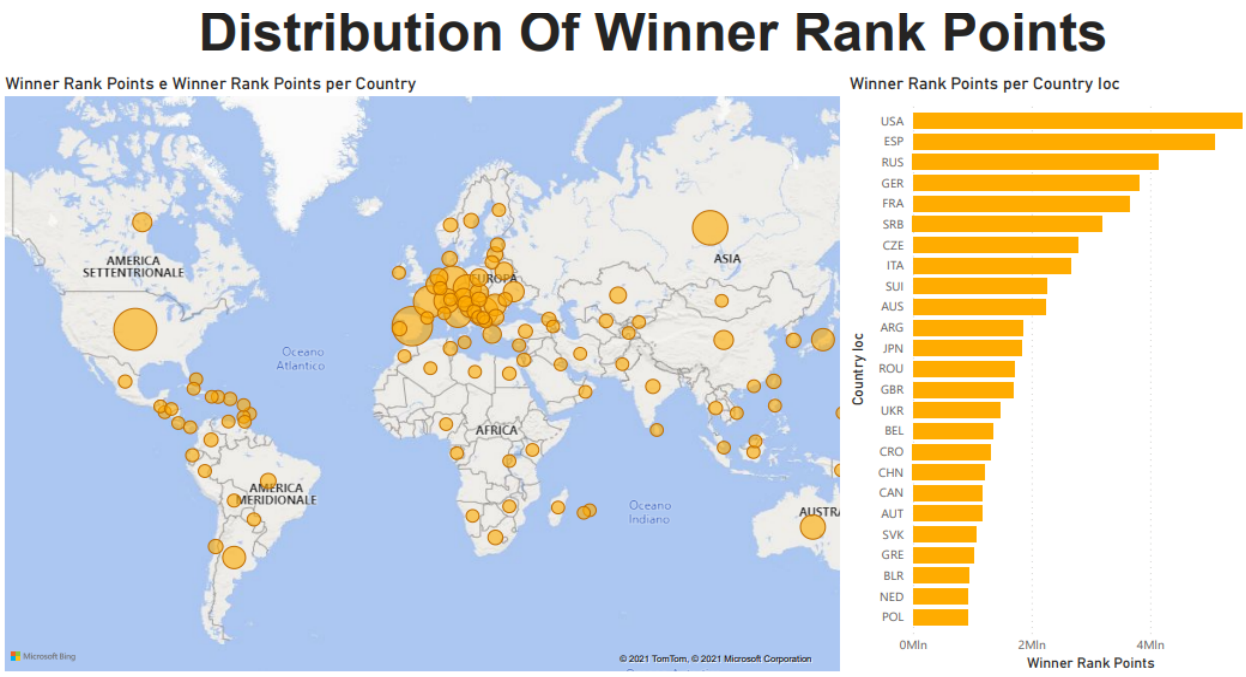
Nella query finale, sulle colonne, viene inserito il set contenente le tre misure AVG_winner_rank_points, Loser_Rank_Points e Ratio. Sulle righe invece per ogni riga, viene generato per ogni torneo, per ogni anno, il giocatore con Ratio più alto.

Di seguito vengono mostrate le prime righe:

			AVG_Winner_rank_points	Loser_Rank_Points	Ratio
Abu Dhabi	2021	Sofia Kenin	1567.1724137931	5760	3.67540925893329
Acapulco	2016	Victoria Azarenka	946.020408163265	2935	3.10247006795383
Acapulco	2017	Kristina Mladenovic	830.632653061224	1580	1.90216456598118
Acapulco	2018	Alexander Zverev	1473.79591836735	4450	3.01941398028138
Acapulco	2019	Rafael Nadal	1477.10204081633	8320	5.63265080549338
Acapulco	2020	Alexander Zverev	1875.65306122449	3885	2.07127857508133
Acapulco	2021	Stefanos Tsitsipas	1702.96610169492	6765	3.97248071659617
Adelaide	2020	Felix Auger Aliassime	999.893617021277	1656	1.65617618895627
Adelaide	2021	Ashleigh Barty	1515.74418604651	9186	6.06038940116912
Agadir $15K	2017	Abir El Fahimi	83.6451612903226	132	1.57809487080602

Assignment 4

Create a dashboard that shows the geographical distribution of winner rank points and loser rank points.



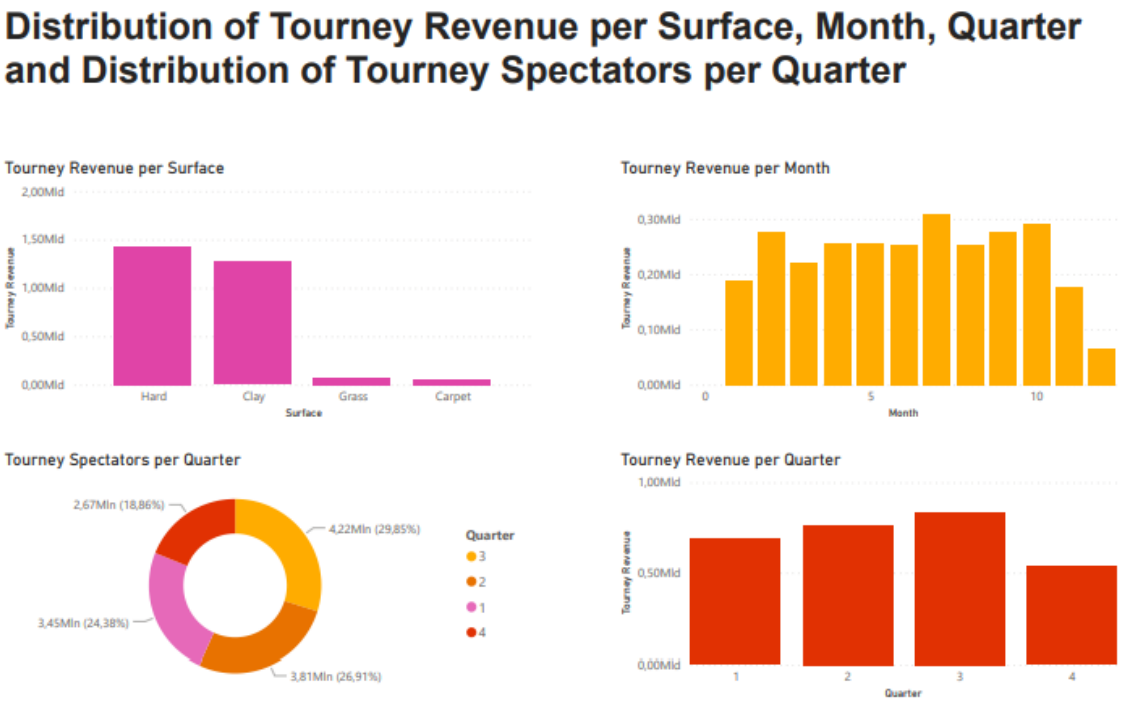
Per svolgere questo assignment è stato utilizzato Power BI. Dopo aver collegato il Software al nostro database,abbiamo rappresentato le distribuzioni geografiche di winner rank point e loser rank point utilizzando due diversi grafici:

- Una prima rappresentazione dei risultati è stata presentata mediante una *distribuzione a bolle* sulla mappa geografica che, attraverso la grandezza delle sfere, è capace di far intuire la quantità di punti per country.
- Una seconda rappresentazione riferisce invece ad un *grafico a barre*, il quale permette di capire le relazioni di grandezza e identificare anche numericamente quale delle

nazioni abbia una distribuzione maggiore rispetto alle altre, creando di fatto una classifica.

Assignment 5

Create a plot/dashboard of your choosing, that you deem interesting w.r.t. the data available in your cube



Anche per quest'ultima richiesta è stato utilizzato il software Power BI. Una delle informazioni che secondo noi può essere interessante conoscere è la seguente:

- Distribuzione di *Tourney_Revenue* per *Surface, Month, Quarter* e inoltre la *Distribuzione Di Spettatori* che assistono alla partita per Quarter.

Entrambe queste informazioni sono raccolte all' interno dell'immagine sopra descritta. Sono stati utilizzati tre diversi bar chart per Surface,Month e Quarter, vista la loro facile comprensibilità mentre,per rappresentare il numero di spettatori per ogni Quarter, è stato utilizzato un grafico a torta.