

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**  
**NÚCLEO DE EDUCAÇÃO A DISTÂNCIA**  
**Pós-graduação *Lato Sensu* em Ciência de Dados e Big Data**

**Gilberto Klingen**

**Determinação de “ODDS” em apostas de jogos de tênis**

Rio de Janeiro  
2021

**Gilberto Klingen**

**Determinação de “ODDS” em apostas de jogos de tênis**

Trabalho de Conclusão de Curso apresentado  
ao Curso de Especialização em Ciência de  
Dados e Big Data como requisito parcial à  
obtenção do título de especialista.

Rio de Janeiro

2021

## SUMÁRIO

<b>1. Introdução.....</b>	<b>4</b>
<b>1.1. Contextualização .....</b>	<b>4</b>
<b>1.2. O problema proposto .....</b>	<b>5</b>
<b>2. Coleta de Dados .....</b>	<b>6</b>
<b>3. Processamento/Tratamento de Dados .....</b>	<b>12</b>
<b>4. Análise e Exploração dos Dados .....</b>	<b>19</b>
<b>5. Criação de Modelos de Machine Learning .....</b>	<b>20</b>
<b>6. Apresentação dos Resultados .....</b>	<b>24</b>
<b>7. Links .....</b>	<b>25</b>
<b>APÊNDICE.....</b>	<b>26</b>

## 1. Introdução

### 1.1. Contextualização

Neste trabalho busquei estudar e analisar um problema em uma área de interesse pessoal. Sou praticante de tênis e acompanho o circuito profissional (pela TV e ao vivo) há muito tempo. Em 2019 decidi adicionar um pouco mais de emoção a esta atividade através de apostas realizadas através de sites específicos (SportingBet, Bet365, BetWay, BetFair, etc).

Estes sites permitem apostas em várias modalidades diferentes tais como: vencedor da partida, vencedor do 1º set, quantidade de games vencidos pelo vencedor da partida, etc. A modalidade que mais me atrai é a mais óbvia de todas : apostar no vencedor da partida.

Para cada aposta realizada existe uma remuneração prevista (ODDS) para cada um dos dois jogadores caso os mesmos vençam a partida. Estes ODDS são calculados levando-se em conta dados de performance de cada um dos jogadores individualmente e na comparação direta entre os adversários (ex: quantos jogos foram realizados entre eles e qual a quantidade de vitórias de cada jogador).

O que mais me intrigou ao analisar os dados de odds dos sites é a performance da previsão média de resultados dos mesmos. Vejam, abaixo, o percentual de acerto de 3 anos consecutivos analisados.

	2017		2018		2019	
	Jogos	%	Jogos	%	Jogos	%
<b>Previsões corretas</b>	1.787	67,87%	1.794	68,03%	1.755	67,24%
<b>Previsões erradas</b>	846	32,13%	843	31,97%	855	32,76%
<b>Total</b>	2.633		2.637		2.610	

O percentual de erro é, em média, de quase 32% para estes 3 anos, o que me fez pensar se seria possível obter melhores resultados através da compilação de dados coletados após cada partida realizada no circuito profissional da ATP.

## 1.2. O problema proposto

Para descrever o problema utilizaremos a técnica dos [5-Ws](#).

**Why?** Minha proposta, um pouco pretenciosa, é tentar desenvolver um modelo de predição de Machine Learning que seja capaz de gerar melhores ODDS e, desta forma, ser possível fazer apostas com uma margem de segurança maior.

**Who?** Os dados analisados virão de duas fontes distintas:

**Resultados dos jogos** : Free ATP and WTA Results and Stats Databases

**ODDS de sites de apostas** : Tennis-Data.co.uk

**What?** Através dos resultados dos jogos iremos montar uma base de dados de performance individual de cada jogador e do par de jogadores para tentarmos montar um modelo que seja apto a nos fornecer melhores ODDS para cada jogador em cada partida a ser disputada.

**Where?** Os dados refletem os resultados de todos os jogos de tênis da ATP (Associação de Tenistas Profissionais) entre dois jogadores (os jogos de dupla não serão analisados) através do mundo.

**When?** Para termos uma base completa de resultados analisados decidi iniciar a partir de 1998 quando foi disputada a primeira partida de ATP do jogador mais velho em atividade no circuito de tênis: Roger Federer. Decidi não incluir o ano de 2020 nas análises pois, com a pandemia do Covid-19, muitos jogadores decidiram não participar dos torneios até que a situação voltasse ao normal. Sendo assim, alguns jogadores teriam a sua performance “melhorada” em função da ausência de jogadores de ponta (Topo do Ranking). O próprio Roger Federer, acima citado, é um exemplo disto (ocupa a quinta posição do ranking e não participou de torneios neste ano).

## 2. Coleta de Dados

### Dataset de Resultado dos jogos

Site: Free ATP and WTA Results and Stats Databases

Link <https://www.tennisabstract.com/blog/2015/03/24/free-atp-and-wta-results-and-stats-databases/>

Link de acesso aos dados por ano:

[https://github.com/JeffSackmann/tennis\\_atp/blob/master/atp\\_matches\\_<ano>.csv](https://github.com/JeffSackmann/tennis_atp/blob/master/atp_matches_<ano>.csv)

Onde <ano> é o ano do qual se quer obter os dados

Layout:

Nome da Coluna	Descrição	Tipo	Observação
<b>tourney_id</b>	Código de identificação do torneio	Texto	
<b>tourney_name</b>	Nome e cidade onde está sendo disputado o torneio.	Texto	
<b>surface</b>	Tipo de superfície (dura, saibro ou grama) (Hard, Clay, Grass)	Texto	
<b>draw_size</b>	Quantidade de jogadores inscritos no torneio.	Inteiro	
<b>tourney_level</b>	Código identificador do nível do torneio.	Texto	
<b>tourney_date</b>	Data de início do torneio.	Texto	Formato : AAAAMMDD (*1)
<b>match_num</b>	Número de identificação da partida no torneio.	Inteiro	
<b>winner_id</b>	Chave identificadora do vencedor.	Inteiro	
<b>winner_seed</b>	Informa se o vencedor é cabeça-de-chave e qual é a numeração.	Inteiro	
<b>winner_entry</b>	Maneira pela qual o vencedor entrou no torneio.	Texto	
<b>winner_name</b>	Nome do vencedor da partida.	Texto	
<b>winner_hand</b>	Informa qual a mão dominante do vencedor (direita ou esquerda).	Texto	
<b>winner_ht</b>	Altura do vencedor (cm).	Inteiro	
<b>winner_ioc</b>	Sigla (Comite Olímpico Internacional) do país de nascença do vencedor.	Texto	
<b>winner_age</b>	Idade do vencedor.	Dupla precisão	Formato : AA,DD (*2)
<b>loser_id</b>	Chave identificadora do perdedor.	Inteiro	

Nome da Coluna	Descrição	Tipo	Observação
<b>loser_seed</b>	Informa se o perdedor é cabeça-de-chave e qual é a numeração.	Inteiro	
<b>loser_entry</b>	Maneira pela qual o perdedor entrou no torneio.	Texto	
<b>loser_name</b>	Nome do perdedor da partida.	Texto	
<b>loser_hand</b>	Informa qual a mão dominante do perdedor (direita ou esquerda).	Texto	
<b>loser_ht</b>	Altura do perdedor (cm).	Inteiro	
<b>loser_ioc</b>	Sigla (Comite Olímpico Internacional) do país de nascença do perdedor.	Texto	
<b>loser_age</b>	Idade do perdedor.	Dupla precisão	Formato : AA,DD (*2)
<b>score</b>	Resultado final da partida (games por set).	Texto	
<b>best_of</b>	Quantidade máxima de sets jogados (melhor de).	Inteiro	
<b>round</b>	Identificação da rodada (1ª, 2ª, ... , final)	Texto	
<b>minutes</b>	Duração da partida em minutos.	Inteiro	
<b>w_ace</b>	Quantidade de aces do vencedor.	Inteiro	
<b>w_df</b>	Quantidade de duplas faltas do vencedor.	Inteiro	
<b>w_svpt</b>	Quantidade de pontos de serviço do vencedor.	Inteiro	
<b>w_1stIn</b>	Quantidade de primeiros serviços acertados pelo vencedor.	Inteiro	
<b>w_1stWon</b>	Quantidade de primeiros serviços vencidos pelo vencedor.	Inteiro	
<b>w_2ndWon</b>	Quantidade de segundos serviços vencidos pelo vencedor.	Inteiro	
<b>w_SvGms</b>	Quantidade de games de serviço do vencedor.	Inteiro	
<b>w_bpSaved</b>	Quantidade de break points salvos pelo vencedor	Inteiro	
<b>w_bpFaced</b>	Quantidade de break points jogados pelo vencedor	Inteiro	
<b>l_ace</b>	Quantidade de aces do perdedor.	Inteiro	
<b>l_df</b>	Quantidade de duplas faltas do perdedor.	Inteiro	
<b>l_svpt</b>	Quantidade de pontos de serviço do perdedor.	Inteiro	
<b>l_1stIn</b>	Quantidade de primeiros serviços acertados pelo perdedor.	Inteiro	
<b>l_1stWon</b>	Quantidade de primeiros serviços vencidos pelo perdedor.	Inteiro	

Nome da Coluna	Descrição	Tipo	Observação
<b>I_2ndWon</b>	Quantidade de segundos serviços vencidos pelo perdedor.	Inteiro	
<b>I_SvGms</b>	Quantidade de games de serviço do perdedor.	Inteiro	
<b>I_bpSaved</b>	Quantidade de break points salvos pelo perdedor	Inteiro	
<b>I_bpFaced</b>	Quantidade de break points jogados pelo perdedor	Inteiro	
<b>winner_rank</b>	Ranking do vencedor.	Inteiro	
<b>winner_rank_points</b>	Quantidade de pontos no ranking do vencedor.	Inteiro	
<b>loser_rank</b>	Ranking do perdedor.	Inteiro	
<b>loser_rank_points</b>	Quantidade de pontos no ranking do perdedor.	Inteiro	

(\*1) A data não exprime o dia exato em que ocorreu a partida mas a data inicial do torneio

(\*2) DD = Decimais correspondentes aos meses de idade.

Domínios de dados:

<b>draw_size</b>	<b>tourney_level</b>		<b>winner_entry</b>		<b>round</b>	
4	A	ATP 250/500	Alt	Alternate	F	Final
8	G	Grand Slam	LL	Lucky Loser	SF	Semi-final
32	M	Master 1000	PR	Protected Ranking	QF	Quartas-de-final
64	F	Finals	Q	Qualifying	R16	Oitavas-de-final
128	D	Davis Cup	SE	Special Exempt	R32	
			WC	Wild Card	R128	
					R64	
					RR	Round Robin



### Dataset de ODDS de sites de apostas

Site: Tennis-Data.co.uk

Link: <http://www.tennis-data.co.uk/alldata.php>

Link de acesso aos dados por ano:

<http://www.tennis-data.co.uk/<ano>/<ano>.zip>

Onde <ano> é o ano do qual se quer obter os dados

Layout:

Nome da Coluna	Descrição	Tipo	Observação
<b>ATP</b>	Número sequencial do torneio da ATP no ano.	Inteiro	
<b>Location</b>	Cidade onde está sendo disputado o torneio.	Texto	
<b>Tournament</b>	Nome do torneio.	Texto	
<b>Date</b>	Data da partida.	Texto	Formato : AAAAMMDD (*1)
<b>Series</b>	Classificação do tipo de torneio da ATP.	Texto	
<b>Court</b>	Informa se a quadra é coberta (Indoor) ou aberta (Outdoor).	Texto	
<b>Surface</b>	Tipo de superfície (dura, saibro ou grama) (Hard, Clay, Grass)	Texto	
<b>Round</b>	Identificação da rodada (1ª, 2ª, ... , final)	Texto	
<b>Best of</b>	Quantidade máxima de sets jogados (melhor de).	Inteiro	
<b>Winner</b>	Nome do vencedor da partida.	Texto	
<b>Loser</b>	Nome do perdedor da partida.	Texto	
<b>WRank</b>	Ranking do vencedor.	Inteiro	
<b>LRank</b>	Ranking do perdedor.	Inteiro	
<b>WPts</b>	Quantidade de pontos no ranking do vencedor.	Inteiro	
<b>LPts</b>	Quantidade de pontos no ranking do perdedor.	Inteiro	
<b>W1</b>	Quantidade games ganhos pelo vencedor no set 1.	Inteiro	
<b>L1</b>	Quantidade games ganhos pelo perdedor no set 1.	Inteiro	
<b>W2</b>	Quantidade games ganhos pelo vencedor no set 2.	Inteiro	

Nome da Coluna	Descrição	Tipo	Observação
<b>L2</b>	Quantidade games ganhos pelo perdedor no set 2.	Inteiro	
<b>W3</b>	Quantidade games ganhos pelo vencedor no set 3.	Inteiro	
<b>L3</b>	Quantidade games ganhos pelo perdedor no set 3.	Inteiro	
<b>W4</b>	Quantidade games ganhos pelo vencedor no set 4.	Inteiro	
<b>L4</b>	Quantidade games ganhos pelo perdedor no set 4.	Inteiro	
<b>W5</b>	Quantidade games ganhos pelo vencedor no set 5.	Inteiro	
<b>L5</b>	Quantidade games ganhos pelo perdedor no set 5.	Inteiro	
<b>Wsets</b>	Quantidade de sets ganhos pelo vencedor.	Inteiro	
<b>Lsets</b>	Quantidade de sets ganhos pelo perdedor.	Inteiro	
<b>Comment</b>	Informa se o jogo foi até o final (Completed) , se houve W.O. (Walkover), se o perdedor se retirou antes do fim da partida (Retired) , se o perdedor foi desclassificado (Awarded)	Texto	
<b>B365W</b>	Probabilidade de vitória atribuída ao vencedor pelo site Bet 365.	Dupla precisão	
<b>B365L</b>	Probabilidade de vitória atribuída ao perdedor pelo site Bet 365.	Dupla precisão	
<b>PSW</b>	Probabilidade de vitória atribuída ao vencedor pelo site Pinnacles Sports.	Dupla precisão	
<b>PSL</b>	Probabilidade de vitória atribuída ao perdedor pelo site Pinnacles Sports.	Dupla precisão	
<b>MaxW</b>	Probabilidade máxima de vitória atribuída ao vencedor por todos os sites de apostas.	Dupla precisão	
<b>MaxL</b>	Probabilidade máxima de vitória atribuída ao perdedor por todos os sites de apostas.	Dupla precisão	
<b>AvgW</b>	Probabilidade média de vitória atribuída ao vencedor por todos os sites de apostas.	Dupla precisão	
<b>AvgL</b>	Probabilidade média de vitória atribuída ao perdedor por todos os sites de apostas.	Dupla precisão	

### Observações:

O layout acima exposto não é único para todos os anos.

Entre 2001 e 2019 foram incluídos e excluídos os odds de alguns sites de apostas. No período de 2001 a 2009 não havia as colunas AvgW (odds médios de todos os sites para o vencedor) e AvgL (odds médios de todos os sites para o perdedor). Elas foram calculadas pelos processos de obtenção dos dados (Python).

### Lista dos sites que entraram e saíram da relação

Sites	
B365	Bet365
B&W	Bet&Win
CB	Centrebet
EX	Expekt
LB	Ladbrokes
GB	Gamebookers
IW	Interwetten
OS	Pinnacles Sports
SB	Sportingbet
SJ	Stan James
UB	Unibet

### Domínios de dados:

Series	Court	Surface	Round
ATP250	Indoor	Hard	1st Round
Grand Slam	Outdoor	Clay	2nd Round
ATP500		Grass	3rd Round
Masters 1000			4th Round
Masters Cup			Quarterfinals
			Semifinals
			The Final
			Round Robin

Best of	Comment
3	Completed
5	Walkover
	Retired
	Awarded
	Sched

### 3. Processamento/Tratamento de Dados

Conforme citado diversas vezes no curso, esta é a etapa mais trabalhosa de todas. Dividi em duas fases : obtenção dos dados (obtenção, limpeza, preparação de dados) e geração da base de dados final (a ser utilizada na fase de modelagem de Machine Learning).

A primeira questão que surge é a junção dos dois datasets. Ambos contém dados de cada partida disputada que são relevantes para a análise do problema e geração dos modelos. A chave de identificação de cada partida seria (torneio, vencedor e perdedor). Em tese, estes três dados seriam suficientes para identificar univocamente uma partida mas, como vimos em um momento já adiantado do processo de preparação de dados, existe um tipo de torneio onde os jogos não são de caráter eliminatório. São os torneios do tipo “Finals” que são realizados ao término da temporada e são disputados entre os 8 melhores jogadores do ano. Este torneio é dividido em 2 fases. A primeira é classificatória com 2 grupos de 4 jogadores no sistema “round robin”, onde todos os jogadores de cada chave jogam entre si. Os dois primeiros classificados de cada grupo disputam uma partida semi-final e os vencedores destas partidas jogarão a final. Desta forma, dois jogadores que se enfrentaram na fase classificatória podem jogar novamente na final. Sendo assim, tive que adicionar mais uma coluna identificando em que rodada (round) foi executada a partida. Minha primeira opção para diferenciar os jogos seria a data mas, conforme disse anteriormente, a data utilizada no dataset de resultados é a do primeiro dia da competição.

Após definidas as colunas para a junção dos datasets percebemos que deveríamos criar tabelas auxiliares de conversão de valores (DE-PARA) pois em cada dataset as colunas vinham preenchidas de forma diferente. Para auxiliar no processo de geração destas tabelas utilizamos programas Python, Planilhas Excel (contendo o resultado dos dados extraídos pelo Python) e consultas ao site da ATP e a Wikipedia, principalmente na geração da tabela de jogadores pois, em alguns casos, os nomes utilizados em cada dataset eram completamente diferentes. Este processo exigiu muito esforço de pesquisa (via Internet) e comparações visuais (não

automatizadas). As funções de Python utilizadas para geração das tabelas de DE-PARA foram :

`gera_lista_jogadores_arquivo_ODDS_em_EXCEL()`

`gera_lista_jogadores_arquivo_results_em_EXCEL()`

`gera_lista_torneios_arquivo_ODDS_em_EXCEL()`

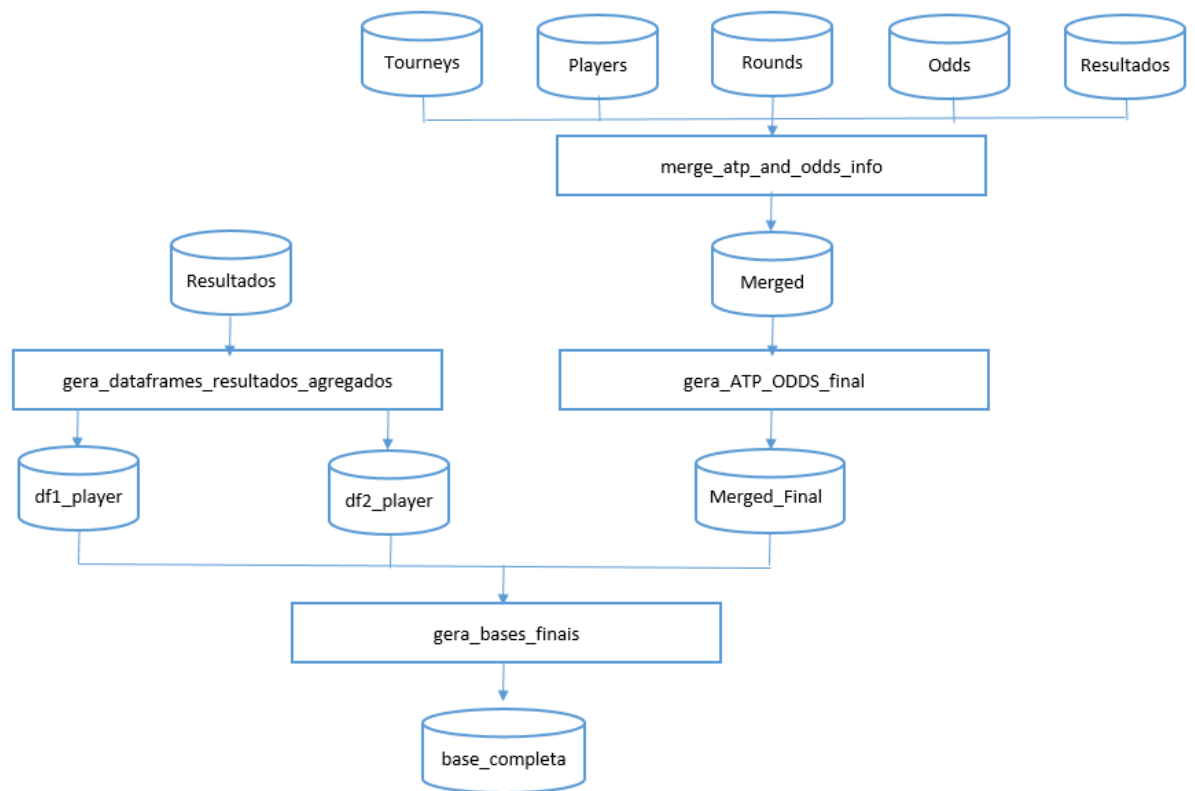
`gera_lista_torneios_arquivo_results_em_EXCEL()`

Após o término da fase de limpeza e preparação foi possível iniciar a fase de geração da base final que será submetida ao modelo de Machine Learning.

Utilizamos programas Python chamando funções que geravam dados que seriam tratados por funções subsequentes até que fosse possível obter a base final.

Segue , abaixo, uma descrição dos passos utilizados para geração dos dados.

Função	Objetivo	Entradas	Saídas
<code>merge_atp_and_odds_info()</code>	Gerar uma base com a junção dos dados de resultados da ATP e ODDS.	players_de_para.xlsx tournaments_de_para.xlsx round_de_para.xlsx dataframe_ODDS(*) dataframe_resultados(*)	merged_ATP_ODDS.xlsx
<code>gera_dataframes_resultados_agregados()</code>	Gerar duas bases. A base df1 conterá todos os dados de performance individual de cada jogador após o término da partida. A base df2 conterá todos os dados de performance do par de jogadores, um contra o outro.	dataframe_resultados(*)	df1_player_geral.csv df2_player_geral.csv
<code>gera_ATP_ODDS_final()</code>	Seleciona somente as colunas relevantes da base merged, adiciona as colunas : winner (1 ou 2) e player_1_id e player_2_id. Ao final remove todas as linhas cujos os ODDS médios sejam nulos pois não serão úteis para a base do modelo.	merged_ATP_ODDS.xlsx	merged_ATP_ODDS_final.csv
<code>gera_bases_finais()</code>	Altera a tabela merged de forma que os dados associados a cada partida sejam os valores agregados antes do início da mesma e não ao término da partida. Desta forma, poderemos estimar quais as ODDS de cada jogador com base nas informações que existiam antes do início da partida (sem contar com o resultado da partida vigente). Faz a junção de dados agregados e ODDS.	merged_ATP_ODDS_final.csv df1_player_geral.csv df2_player_geral.csv	base_completa.xlsx



### Dataset da base completa para modelagem

Nome da Coluna	Descrição	Tipo	Observação
<b>tourney_id</b>	Código de identificação do torneio	Texto	
<b>tourney_date</b>	Data de início do torneio.	Texto	Formato : AAAAMMDD
<b>match_num</b>	Número de identificação da partida no torneio.	Inteiro	
<b>round</b>	Identificação da rodada (1ª, 2ª, ..., final)	Texto	
<b>surface</b>	Tipo de superfície (dura, saibro ou grama) (Hard, Clay, Grass)	Texto	
<b>tourney_level</b>	Código identificador do nível do torneio.	Texto	
<b>best_of</b>	Quantidade máxima de sets jogados (melhor de).	Inteiro	
<b>AvgW</b>	Probabilidade média de vitória atribuída ao vencedor por todos os sites de apostas.	Dupla precisão	
<b>AvgL</b>	Probabilidade média de vitória atribuída ao perdedor por todos os sites de apostas.	Dupla precisão	

Nome da Coluna	Descrição	Tipo	Observação
<b>player_id_1</b>	Chave identificadora do jogador 1 (*)	Inteiro	
<b>pct_surface_victories_1</b>	Percentual de vitórias do jogador 1 na superfície em que está sendo disputada a partida.	Dupla precisão	
<b>pct_best_of_victories_1</b>	Percentual de vitórias do jogador 1 na quantidade de sets (3 ou 5) em que está sendo disputada a partida.	Dupla precisão	
<b>avg_ace_1</b>	Quantidade média de aces do jogador 1 na data da partida.	Dupla precisão	
<b>avg_df_1</b>	Quantidade média de duplas faltas do jogador 1 na data da partida.	Dupla precisão	
<b>avg_svpt_1</b>	Quantidade média de pontos de serviço do jogador 1 na data da partida.	Dupla precisão	
<b>avg_1stIn_1</b>	Quantidade média de primeiros serviços acertados pelo jogador 1 na data da partida.	Dupla precisão	
<b>avg_1stWon_1</b>	Quantidade média de primeiros serviços vencidos pelo jogador 1 na data da partida.	Dupla precisão	
<b>avg_2ndWon_1</b>	Quantidade média de segundos serviços vencidos pelo jogador 1 na data da partida.	Dupla precisão	
<b>avg_SvGms_1</b>	Quantidade média de games de serviço do jogador 1 na data da partida.	Dupla precisão	
<b>avg_bpSaved_1</b>	Quantidade média de break points salvos pelo jogador 1 na data da partida.	Dupla precisão	
<b>avg_bpFaced_1</b>	Quantidade média de break points jogados pelo jogador 1 na data da partida.	Dupla precisão	
<b>age_1</b>	Idade do jogador 1 na data da partida.	Inteiro	
<b>rank_points_1</b>	Quantidade de pontos no ranking do jogador 1 na data da partida.	Inteiro	
<b>pct_victories_grand_slam_1</b>	Percentual de vitórias do jogador 1 em jogos de Grand Slam na data da partida.	Dupla precisão	

Nome da Coluna	Descrição	Tipo	Observação
<b>pct_victories_master_1000_1</b>	Percentual de vitórias do jogador 1 em jogos de Master 1000 na data da partida.	Dupla precisão	
<b>pct_victories_ATP_250_500_1</b>	Percentual de vitórias do jogador 1 em jogos de ATP 250 ou 500 na data da partida.	Dupla precisão	
<b>pct_victories_finals_1</b>	Percentual de vitórias do jogador 1 em jogos de Finals na data da partida.	Dupla precisão	
<b>qty_grand_slam_titles_1</b>	Quantidade de títulos de Grand Slam obtidos pelo jogador 1 na data da partida.	Inteiro	
<b>qty_master_1000_titles_1</b>	Quantidade de títulos de Master 1000 obtidos pelo jogador 1 na data da partida.	Inteiro	
<b>qty_ATP_250_500_titles_1</b>	Quantidade de títulos de ATP 250 ou 500 obtidos pelo jogador 1 na data da partida.	Inteiro	
<b>qty_finals_titles_1</b>	Quantidade de títulos de Finals obtidos pelo jogador 1 na data da partida.	Inteiro	
<b>player_id_2</b>	Chave identificadora do jogador 2(*)	Inteiro	
<b>pct_surface_victories_2</b>	Percentual de vitórias do jogador 2 na superfície em que está sendo disputada a partida na data da partida.	Dupla precisão	
<b>pct_best_of_victories_2</b>	Percentual de vitórias do jogador 2 na quantidade de sets (3 ou 5) em que está sendo disputada a partida na data da partida.	Dupla precisão	
<b>avg_ace_2</b>	Quantidade média de aces do jogador 2 na data da partida.	Dupla precisão	
<b>avg_df_2</b>	Quantidade média de duplas faltas do jogador 2 na data da partida.	Dupla precisão	
<b>avg_svpt_2</b>	Quantidade média de pontos de serviço do jogador 2 na data da partida.	Dupla precisão	
<b>avg_1stIn_2</b>	Quantidade média de primeiros serviços acertados pelo jogador 2 na data da partida.	Dupla precisão	



Nome da Coluna	Descrição	Tipo	Observação
<b>avg_1stWon_2</b>	Quantidade média de primeiros serviços vencidos pelo jogador 2 na data da partida.	Dupla precisão	
<b>avg_2ndWon_2</b>	Quantidade média de segundos serviços vencidos pelo jogador 2 na data da partida.	Dupla precisão	
<b>avg_SvGms_2</b>	Quantidade média de games de serviço do jogador 2 na data da partida.	Dupla precisão	
<b>avg_bpSaved_2</b>	Quantidade média de break points salvos pelo jogador 2 na data da partida.	Dupla precisão	
<b>avg_bpFaced_2</b>	Quantidade média de break points jogados pelo jogador 2 na data da partida.	Dupla precisão	
<b>age_2</b>	Idade do jogador 2 na data da partida.	Inteiro	
<b>rank_points_2</b>	Quantidade de pontos no ranking do jogador 2 na data da partida.	Inteiro	
<b>pct_victories_grand_slam_2</b>	Percentual de vitórias do jogador 2 em jogos de Grand Slam na data da partida.	Dupla precisão	
<b>pct_victories_master_1000_2</b>	Percentual de vitórias do jogador 2 em jogos de Master 1000 na data da partida.	Dupla precisão	
<b>pct_victories_ATP_250_500_2</b>	Percentual de vitórias do jogador 2 em jogos de ATP 250 ou 500 na data da partida.	Dupla precisão	
<b>pct_victories_finals_2</b>	Percentual de vitórias do jogador 2 em jogos de Finals na data da partida.	Dupla precisão	
<b>qty_ATP_250_500_titles_2</b>	Quantidade de títulos de ATP 250 ou 500 obtidos pelo jogador 2 na data da partida.	Inteiro	
<b>qty_grand_slam_titles_2</b>	Quantidade de títulos de Grand Slam obtidos pelo jogador 2 na data da partida.	Inteiro	
<b>qty_master_1000_titles_2</b>	Quantidade de títulos de Master 1000 obtidos pelo jogador 2 na data da partida.	Inteiro	
<b>qty_finals_titles_2</b>	Quantidade de títulos de Finals obtidos pelo jogador 2 na data da partida.	Inteiro	

Nome da Coluna	Descrição	Tipo	Observação
<b>pct_player_1_surface_victories</b>	Percentual de vitórias do jogador 1 sobre o jogador 2 na superfície em que está sendo disputada a partida.	Dupla precisão	
<b>pct_player_1_best_of_victories</b>	Percentual de vitórias do jogador 1 sobre o jogador 2 na quantidade de sets (3 ou 5) em que está sendo disputada a partida na data da partida.	Dupla precisão	
<b>pct_player_1_grand_slam_victories</b>	Percentual de vitórias do jogador 1 sobre o jogador 2 em jogos de Grand Slam na data da partida.	Dupla precisão	
<b>pct_player_1_master_1000_victories</b>	Percentual de vitórias do jogador 1 sobre o jogador 2 em jogos de Master 1000 na data da partida.	Dupla precisão	
<b>pct_player_1_finals_victories</b>	Percentual de vitórias do jogador 1 sobre o jogador 2 em jogos de Finals na data da partida.	Dupla precisão	
<b>pct_player_1_ATP_250_500_victories</b>	Percentual de vitórias do jogador 1 sobre o jogador 2 em jogos de ATP 250 ou 500 na data da partida.	Dupla precisão	
<b>pct_player_2_surface_victories</b>	Percentual de vitórias do jogador 2 sobre o jogador 1 na superfície em que está sendo disputada a partida na data da partida.	Dupla precisão	
<b>pct_player_2_best_of_victories</b>	Percentual de vitórias do jogador 2 sobre o jogador 1 na quantidade de sets (3 ou 5) em que está sendo disputada a partida na data da partida.	Dupla precisão	
<b>pct_player_2_grand_slam_victories</b>	Percentual de vitórias do jogador 2 sobre o jogador 1 em jogos de Grand Slam na data da partida.	Dupla precisão	
<b>pct_player_2_master_1000_victories</b>	Percentual de vitórias do jogador 2 sobre o jogador 1 em jogos de Master 1000 na data da partida.	Dupla precisão	

Nome da Coluna	Descrição	Tipo	Observação
<b>pct_player_2_finals_victories</b>	Percentual de vitórias do jogador 2 sobre o jogador 1 em jogos de Finals na data da partida.	Dupla precisão	
<b>pct_player_2_ATP_250_500_victories</b>	Percentual de vitórias do jogador 2 sobre o jogador 1 em jogos de ATP 250 ou 500 na data da partida.	Dupla precisão	
<b>winner</b>	Número do jogador que venceu a partida (1 ou 2).	Inteiro	

### Observações :

- 1) Quando dizemos que o dado foi colhido na data da partida estamos nos referindo a dados anteriores ao início da partida naquela data.
- 2) A escolha de jogador 1 e 2 é feita da seguinte forma: o jogador de menor player\_id será o jogador 1 e outro será o jogador 2.
- 3) Retiramos da base todas as partidas de Copa Davis pois não existem odds associados a elas.
- 4) Retiramos da base a coluna altura do jogador pois grande parte delas estava em branco.
- 5) A quantidade final de linhas disponíveis é de 44.668

## 4. Análise e Exploração dos Dados

Na verdade, toda a análise relevante de dados foi feita a priori pois foi justamente através dela que surgiu a idéia para a realização deste trabalho. Fizemos nossas análises preliminares utilizando a base de ODDS de 2017 a 2019.

Podemos ver que se estendermos nossa análise para todos os anos existentes na base de modelagem teremos:

	OK	% OK	NOK	% NOK	Total
<b>2001</b>	1.703	67,39%	824	32,61%	2.527
<b>2002</b>	1.803	67,55%	866	32,45%	2.669

	OK	% OK	NOK	% NOK	Total
2003	1.934	69,34%	855	30,66%	2.789
2004	1.897	67,80%	901	32,20%	2.798
2005	2.030	70,86%	835	29,14%	2.865
2006	1.954	68,78%	887	31,22%	2.841
2007	1.894	71,58%	752	28,42%	2.646
2008	1.720	70,75%	711	29,25%	2.431
2009	1.730	70,24%	733	29,76%	2.463
2010	908	71,55%	361	28,45%	1.269
2011	1.480	72,91%	550	27,09%	2.030
2012	1.335	74,41%	459	25,59%	1.794
2013	1.253	71,03%	511	28,97%	1.764
2014	1.245	71,76%	490	28,24%	1.735
2015	1.239	73,01%	458	26,99%	1.697
2016	1.843	71,02%	752	28,98%	2.595
2017	1.758	68,03%	826	31,97%	2.584
2018	1.759	68,10%	824	31,90%	2.583
2019	1.742	67,31%	846	32,69%	2.588

Média	Total
29,82%	44.668

A média geral de ODDS “errados” ainda está na casa dos 30%.

## 5. Criação de Modelos de Machine Learning

Como nosso objetivo é obter um modelo em que para cada partida seja possível obter a probabilidade de vitória do jogador 1 e do jogador 2 e assim conseguirmos calcular os seus respectivos odds, resolvi fazê-lo através de 3 algoritmos propícios para tal, **Logistic Regression**, **Decision Tree Classifier** e **Naive Bayes**. Infelizmente, ao iniciarmos o processo de treinamento do modelo constatamos que havia algo que impossibilitava a execução correta dos cálculos pois o modelo sempre encontrava o jogador “correto” e atingia 100% de acertos.

Por sorte, nossas suspeitas sobre os possíveis motivos que levavam a este erro estavam corretas: precisávamos normalizar os valores presentes na base (somente valores entre 0 e 1,0) e, ao invés de colocar valores de performance atribuídos ao jogador 1 e jogador 2, teríamos os mesmos conceitos expressos nas

diferenças entre esses valores (ex: **pct\_player\_1\_grand\_slam\_victories** e **pct\_player\_2\_grand\_slam\_victories** seriam consolidados em uma única coluna contendo a diferença entre esses valores. **pct\_player\_1\_grand\_slam\_victories = pct\_player\_1\_grand\_slam\_victories - pct\_player\_2\_grand\_slam\_victories**).

Após termos feito essa alteração foi possível seguir com a etapa de modelagem conforme explicado abaixo. O modelo que se mostrou ser o mais indicado para esta tarefa foi o **Logistic Regression**. Seguem os passos executados.

```
import pandas as pd
import numpy as np
from sklearn import datasets, tree
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.feature_extraction import DictVectorizer
from sklearn.preprocessing import LabelEncoder

def obter_base_reduzida(base):

    # Esta função transforma duas colunas de mesmo significado em apenas uma contendo a diferença do valor relativo
    # ao jogador 1 diminuída do valor relativo ao jogador 2

    base['pct_surface_victories_1'] = base['pct_surface_victories_1'] - base['pct_surface_victories_2']
    base['pct_best_of_victories_1'] = base['pct_best_of_victories_1'] - base['pct_best_of_victories_2']
    base['avg_ace_1'] = base['avg_ace_1'] - base['avg_ace_2']
    base['avg_df_1'] = base['avg_df_1'] - base['avg_df_2']
    base['avg_svpt_1'] = base['avg_svpt_1'] - base['avg_svpt_2']
    base['avg_1stIn_1'] = base['avg_1stIn_1'] - base['avg_1stIn_2']
    base['avg_1stWon_1'] = base['avg_1stWon_1'] - base['avg_1stWon_2']
    base['avg_2ndWon_1'] = base['avg_2ndWon_1'] - base['avg_2ndWon_2']
    base['avg_SvGms_1'] = base['avg_SvGms_1'] - base['avg_SvGms_2']
    base['avg_bpSaved_1'] = base['avg_bpSaved_1'] - base['avg_bpSaved_2']
    base['avg_bpFaced_1'] = base['avg_bpFaced_1'] - base['avg_bpFaced_2']
    base['age_1'] = base['age_1'] - base['age_2']
    base['rank_points_1'] = base['rank_points_1'] - base['rank_points_2']
    base['pct_victories_grand_slam_1'] = base['pct_victories_grand_slam_1'] - base['pct_victories_grand_slam_2']
    base['pct_victories_master_1000_1'] = base['pct_victories_master_1000_1'] - base['pct_victories_master_1000_2']
    base['pct_victories_ATP_250_500_1'] = base['pct_victories_ATP_250_500_1'] - base['pct_victories_ATP_250_500_2']
    base['pct_victories_finals_1'] = base['pct_victories_finals_1'] - base['pct_victories_finals_2']
    base['qty_grand_slam_titles_1'] = base['qty_grand_slam_titles_1'] - base['qty_grand_slam_titles_2']
    base['qty_master_1000_titles_1'] = base['qty_master_1000_titles_1'] - base['qty_master_1000_titles_2']
    base['qty_ATP_250_500_titles_1'] = base['qty_ATP_250_500_titles_1'] - base['qty_ATP_250_500_titles_2']
    base['qty_finals_titles_1'] = base['qty_finals_titles_1'] - base['qty_finals_titles_2']

    base['pct_player_1_surface_victories'] = base['pct_player_1_surface_victories'] - base['pct_player_2_surface_victories']
    base['pct_player_1_best_of_victories'] = base['pct_player_1_best_of_victories'] - base['pct_player_2_best_of_victories']
    base['pct_player_1_grand_slam_victories'] = base['pct_player_1_grand_slam_victories'] - base['pct_player_2_grand_slam_victories']
    base['pct_player_1_master_1000_victories'] = base['pct_player_1_master_1000_victories'] - base['pct_player_2_master_1000_victories']
    base['pct_player_1_finals_victories'] = base['pct_player_1_finals_victories'] - base['pct_player_2_finals_victories']
    base['pct_player_1_ATP_250_500_victories'] = base['pct_player_1_ATP_250_500_victories'] - base['pct_player_2_ATP_250_500_victories']

    base = base.drop(['pct_surface_victories_2', 'pct_best_of_victories_2', 'avg_ace_2', 'avg_df_2', 'avg_svpt_2', 'avg_1stIn_2',
                     'avg_1stWon_2', 'avg_2ndWon_2', 'avg_SvGms_2', 'avg_bpSaved_2', 'avg_bpFaced_2', 'age_2', 'rank_points_2',
                     'pct_victories_grand_slam_2', 'pct_victories_master_1000_2', 'pct_victories_ATP_250_500_2',
                     'pct_victories_finals_2', 'qty_grand_slam_titles_2', 'qty_master_1000_titles_2', 'qty_ATP_250_500_titles_2',
                     'qty_finals_titles_2', 'pct_player_2_surface_victories', 'pct_player_2_best_of_victories',
                     'pct_player_2_grand_slam_victories', 'pct_player_2_master_1000_victories', 'pct_player_2_finals_victories',
                     'pct_player_2_ATP_250_500_victories'], axis=1)

    return base

# Obtem a base completa de jogos para análise

base_completa = pd.read_excel(r'C:\Giba\Dados\A_Giba\Documentos\Adm\PUC_MG\Python\Bases_ML\base_completa.xlsx')
```

```
# Gera a base completa redefinida, onde os valores irão de 0,0 a 1,0 e exclui as colunas não relevantes

base_redefinida = base_completa.drop(['tourney_id', 'tourney_date', 'match_num', 'round', 'surface', 'tourney_level',
                                     'best_of', 'AvgW', 'AvgL', 'player_id_1', 'player_id_2'], axis=1)
base_redefinida = base_redefinida.apply(lambda x: round(x / 100, 4))
base_redefinida['rank_points_1'] = base_redefinida['rank_points_1'].apply(lambda x: round(x / 1000, 4))
base_redefinida['rank_points_2'] = base_redefinida['rank_points_2'].apply(lambda x: round(x / 1000, 4))
base_redefinida['winner'] = base_redefinida['winner'].apply(lambda x: x * 100)
base_redefinida.to_excel(r'C:\Giba\Dados\A_Giba\Documentos\Adm\PUC_MG\Python\Bases_ML\base_redefinida.xlsx')
```

```
# Gera a base completa reduzida, onde as colunas do jogador 1 e 2 foram reduzidas a apenas uma contendo a diferença de seus valores
```

```
base_ML = obter_base_reduzida(base_redefinida)
```

## Gera o Modelo de Machine Learning utilizando o algoritmo de regressão logística

```
from sklearn.linear_model import LogisticRegression
```

```
# Separa as colunas de dimensões da base
```

```
X = base_ML.iloc[:, 0:(base_ML.shape[1] - 1)]
```

```
# Separa a coluna de classe de classificação da base
```

```
le = LabelEncoder()
y = le.fit_transform(base_ML.iloc[:, (base_ML.shape[1] - 1)])
```

```
# Faz a separação das bases de treinamento e teste (20% da base será para treinamento)
```

```
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=0)
```

```
# Executa o treinamento do modelo
```

```
LogisticRegr = LogisticRegression(max_iter=1000)
LogisticRegr.fit(x_train, y_train)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=1000,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

```
# Aplica o modelo na base de teste
```

```
y_pred = LogisticRegr.predict(x_test)
```

```
# Verifica a acurácia do modelo
```

```
score = accuracy_score(y_test, y_pred)
print(score)
```

```
0.7799417953884038
```

```
# Verifica a performance de classificação do modelo
```

```
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.76	0.81	0.78	4433
1	0.80	0.75	0.77	4501
accuracy			0.78	8934
macro avg	0.78	0.78	0.78	8934
weighted avg	0.78	0.78	0.78	8934

```
# Verifica a matriz de confusão do modelo
```

```
confusion = confusion_matrix(y_test, y_pred)
cnf_table = pd.DataFrame(data=confusion, index=["Player_1", "Player_2"], columns=["Player_1 (prev)", "Player_2 (prev)"])
print(cnf_table)
```

```
      Player_1 (prev)  Player_2 (prev)
Player_1           3585             848
Player_2           1118            3383
```

```
# Gera a matriz de probabilidades da base completa
```

```
prb = LogisticRegression(random_state=0,max_iter=1000).fit(X, y)
prb.predict_proba(X)
```

```
array([[0.04522779, 0.95477221],
       [0.65610069, 0.34389931],
       [0.69694679, 0.30305321],
       ...,
       [0.59641974, 0.40358026],
       [0.32990118, 0.67009882],
       [0.70340496, 0.29659504]])
```

```
# Salva a matriz de probabilidades em uma planilha Excel
```

```
df_odds = pd.DataFrame(prb.predict_proba(X))
df_odds.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\Bases_ML\\odds_predict_logistic_regression.xlsx")
```

## 6. Apresentação dos Resultados

Data Science WorkFlow Canvas		
Problem Statement	Outcomes/Predictions	Data Acquisition
Tentar obter melhores previsões sobre os possíveis vencedores de partidas de tênis masculinas do circuito profissional (ATP) para utilizá-las em sites de apostas.	Meu objetivo é obter margens de erro inferiores aos atuais 30% médios gerados por alguns sites de apostas.	<p>Obterei dados de duas fontes distintas:</p> <p>1) Resultados dos jogos da ATP desde 1998. (Free ATP and WTA Results and Stats Databases)</p> <p>2) Odds de cada jogadores em partidas da ATP desde 2001. (Tennis-Data.co.uk)</p> <p>Ambos tem milhares de linhas de resultados e odds para cada ano analisado.</p>
Modeling	Data Evaluation	Data Preparation
O modelo mais apropriados para esse tipo de classificação é o de <b>regressão logística</b> . Ao final do processo ele me dará o percentual de probabilidade de vitória de cada jogador.	A performance do modelo será avaliada em função da acurácia de predição que o mesmo apresente ao término desta fase.	<p>Para podermos gerar a base final que será submetida ao modelo deveremos fazer :</p> <p>1) junção das bases de resultados e odds através da chave (torneio, vencedor, perdedor, rodada)</p> <p>2) geração de uma base agregada de dados de performance por cada jogador e para o par de jogadores após cada partida.</p> <p>3) geração da base final onde os valores são associados a cada partida porém antes de sua execução ou seja, com os dados relativos ao final da partida anterior de cada jogador e do par de jogadores.</p>

### Conclusão :

Ao final do processo de predição do modelo conseguimos obter um algoritmo que nos dá 78% de acurácia contra os 70% que os sites de apostas conseguem atingir. Podemos concluir que quanto mais dados de qualidade tivermos acerca do negócio modelado maior será a probabilidade de fazermos previsões mais precisas.



## 7. Links

Link para o vídeo

Parte 1: <https://youtu.be/FdkgggexqAo>

Parte 2: <https://youtu.be/IMUQcUOG0w8>

Link para o repositório:

[https://github.com/Gilbert-Klingen/PUC\\_MG.git](https://github.com/Gilbert-Klingen/PUC_MG.git)

## APÊNDICE

### Programação/Scripts

**TCC PUC-MG Ciencia-de-Dados-e-Big-Data 2019 Gilberto Klingen - Parte 1 -  
Obtenção, limpeza e preparação dos dados.ipynb**

```
#!/usr/bin/env python
```

```
# coding: utf-8
```

```
# ## PUC - MG
```

```
# ## Ciência de Dados e Big Data - 2019
```

```
# ## TCC - Gilberto Klingen
```

```
# # Fontes de Dados :
```

```
#
```

```
# Resultados dos jogos : Free ATP and WTA Results and Stats Databases
```

```
#
```

```
#      https://www.tennisabstract.com/blog/2015/03/24/free-atp-and-wta-results-and-stats-databases/
```

```
#
```

```
# Probabilidade de vitória de cada jogador em cada partida segundo alguns sites de  
apostas : Tennis-Data.co.uk
```

```
#
```

```
# http://www.tennis-data.co.uk/alldata.php
```

```
#
```

```
#
```

```
#
```

```
# ## Aquisição de Dados
```

```
import requests
```

```
import pandas as pd
```

```

import numpy as np
import xlrd
from datetime import datetime, date
from io import StringIO
from IPython.display import display
from zipfile import ZipFile
import os
from bs4 import BeautifulSoup
import wget
import warnings

#
# # Funções

def get_file(url:str, save_path:str, file_name:str):
    r = requests.get(url)

    with open(save_path + file_name, 'wb') as f:
        f.write(r.content)

def unzip_file (filename:str, directory:str):
    with ZipFile(filename, 'r') as zip_ref:
        zip_ref.extractall(directory)

def wget_file(url:str):
    file = wget.download(url)
    print(file)

def obtem_arquivos_dos_links(url:str) -> list:
    lista_arquivos = []

```

```

html_content = requests.get(url).text
soup = BeautifulSoup(html_content, "lxml")
for link in soup.find_all("a"):
    if (format(link.get("href"))[-4:] == 'csv' or format(link.get("href"))[-3:] == 'zip'):
        lista_arquivos.append(format(link.get("href")))
return lista_arquivos

```

```

def obtem_arquivos_dos_urls(url:str) -> list:
    lista_arquivos = []
    html_content = requests.get(url).text
    soup = BeautifulSoup(html_content, "lxml")
    for link in soup.find_all('td id'):
        lista_arquivos.append(format(link.get('class="blob-code blob-code-inner js-file-line"')))
    return lista_arquivos

```

```

def grava_dataframe_retorno_url_2(conteudo_arquivo:str) -> pd.DataFrame:
    dfdata = StringIO(conteudo_arquivo)
    df = pd.read_csv(dfdata, sep=";")
    # print(df)
    # display(df)
    return df

```

```

def          grava_dataframe_retorno_url(conteudo_arquivo:pd.DataFrame)          ->
pd.DataFrame:
    dfdata = conteudo_arquivo[2]
    df = pd.read_csv(dfdata, sep=";")
    # print(df)
    # display(df)
    return df

```

```

def obtem_nome_abreviado_text(nome:str) -> str:
    # de , di, van der, van

    partes_nome=nome.split(' ')

    if 'di' in partes_nome:
        separator_index = partes_nome.index('di')
    elif 'de' in partes_nome:
        separator_index = partes_nome.index('de')
    elif 'van' in partes_nome:
        separator_index = partes_nome.index('van')
    else:
        separator_index = partes_nome.size

    nome = ''
    for ind in range(0,separator_index):
        nome = nome + ' ' + partes_nome[ind][0] + '.'
    nome = nome.strip()

    sobrenome = ' '.join(partes_nome[separator_index:])

    return sobrenome + ' ' + nome

```

```

def obtem_nome_abreviado(lst_nome:list) -> list:
    # de , di, van der, van

    lst_return = []

    for nome in lst_nome:
        partes_nome=str(nome).split(' ')

        if 'di' in partes_nome:

```

```

        separator_index = partes_nome.index('di')
    elif 'de' in partes_nome:
        separator_index = partes_nome.index('de')
    elif 'van' in partes_nome:
        separator_index = partes_nome.index('van')
    else:
        separator_index = len(partes_nome) - 1

    nome = ""
    for ind in range(0,separator_index):
        nome = nome + ' ' + partes_nome[ind][0] + '.'
    nome = nome.strip()

    sobrenome = ' '.join(partes_nome[separator_index:])

    lst_return.append(sobrenome + ' ' + nome)

return lst_return

```

```
def insere_colunas_odds_medios(df_odds:pd.DataFrame):
```

```

    lst_sites = ['CB','B365','GB','EX','IW','LB','SB','PS','SJ','B&W','UB']
    lst_w_sites = []
    lst_l_sites = []

    # Adiciona as colunas de odds medios (AvgW e AvgL)

    df_odds['AvgW'] = np.nan
    df_odds['AvgL'] = np.nan

    # Monta lista de colunas existentes para o ano

    for site in lst_sites:

```

```

column = site + 'W'
if column in df_odds:
    lst_w_sites.append(column)

```

```

column = site + 'L'
if column in df_odds:
    lst_l_sites.append(column)

```

# Calcula os odds medios

```

for ind in df_odds.index:

```

```

    count_w = 0
    count_l = 0

```

```

    sum_w = 0.0
    sum_l = 0.0

```

```

    for site in lst_w_sites:

```

```

        # site_l = site[0:-1] + 'L'

```

```

        # if df_odds.loc[ind,site] > df_odds.loc[ind,site_l]:
        #     # print('{} / {}'.format(df_odds.loc[ind,site], df_odds.loc[ind,site_l]))
        #     aux_swap = df_odds.loc[ind,site]
        #     df_odds.iloc[ind, df_odds.columns.get_loc(site)] = df_odds.loc[ind,site_l]
        #     df_odds.iloc[ind, df_odds.columns.get_loc(site_l)] = aux_swap

```

```

        if not np.isnan(df_odds.loc[ind,site]):
            count_w = count_w + 1
            sum_w = sum_w + df_odds.loc[ind,site]

```

```

    for site in lst_l_sites:

```

```

        if not np.isnan(df_odds.loc[ind,site]):

```

```

        count_l = count_l + 1
        sum_l = sum_l + df_odds.loc[ind,site]

    if count_w != 0:
        df_odds.iloc[ind, df_odds.columns.get_loc('AvgW')] = round(sum_w /
count_w,2)
    if count_l != 0:
        df_odds.iloc[ind, df_odds.columns.get_loc('AvgL')] = round(sum_l / count_l,2)

def obtem_dataframe_resultados_bkp(ano:str) -> pd.DataFrame:

URL="https://github.com/JeffSackmann/tennis_atp/blob/master/atp_matches_{}.csv".f
ormat(ano)

# Obtem o HTML relativo a URL
html_content = requests.get(URL).text

# Obtem as tabelas contidas no HTML
html_tables = pd.read_html(html_content)

# Gera um dataframe contendo os dados de resultados
df0 = html_tables[0]

# Gera um dataframe eliminando a coluna 0 (NaN) e renomeando a coluna 1 para
"Texto"
df1=df0.iloc[:,1:2].rename(columns={1:'Texto'})

# Gera um dataframe transformando a coluna "Texto" em colunas deste novo
dataframe
df2 = df1.Texto.str.split(",",expand=True)

# Gera uma listas contendo os nomes das colunas que estão na linha 0 do
dataframe df2

```



```

lst_col=df2.iloc[0].tolist()
# print(lst_col)

# Gera um dataframe eliminando a primeira linha de df2 (lista de nomes das
colunas)
df3 = df2[1:]

# Renomeia as colunas de df3 de acordo com a lista (lst_col) obtida anteriormente
df3.columns = lst_col

df3['winner_name']= df3['winner_name'].str.strip()
df3['loser_name']= df3['loser_name'].str.strip()

# adiciona a coluna Year (Ano)
df3['Year']=int(ano)

# altera o tipoda coluna match_num para inteiro
df3['match_num'] = df3['match_num'].astype(int)

# ordena o dataframe por data e identificação do jogo
df3.sort_values(['tourney_date','match_num'],inplace=True)
# print(df3)

return df3

def obtem_dataframe_resultados(ano:str) -> pd.DataFrame:

URL="https://github.com/JeffSackmann/tennis_atp/blob/master/atp_matches_{}.csv".f
ormat(ano)

# Obtem o HTML relativo a URL

```

```

html_content = requests.get(URL).text

# Obtem as tabelas contidas no HTML
html_tables = pd.read_html(html_content)

# Gera um dataframe contendo os dados de resultados
df0 = html_tables[0]

if ano == '2020':
    df0.drop("Unnamed: 0", axis=1, inplace=True)
    df3 = df0
else:

    # Gera um dataframe eliminando a coluna 0 (NaN) e renomeando a coluna 1 para
    "Texto"
    df1=df0.iloc[:,1:2].rename(columns={1:'Texto'})

    # Gera um dataframe transformando a coluna "Texto" em colunas deste novo
    dataframe
    df2 = df1.Texto.str.split(" ",expand=True)

    # Gera uma listas contendo os nomes das colunas que estão na linha 0 do
    dataframe df2
    lst_col=df2.iloc[0].tolist()
    # print(lst_col)

    # Gera um dataframe eliminando a primeira linha de df2 (lista de nomes das
    colunas)
    df3 = df2[1:]

    # Renomeia as colunas de df3 de acordo com a lista (lst_col) obtida anteriormente
    df3.columns = lst_col

df3['winner_name']= df3['winner_name'].str.strip()

```

```

df3['loser_name']= df3['loser_name'].str.strip()

# adiciona a coluna Year (Ano)
df3['Year']=int(ano)

# altera o tipoda coluna match_num para inteiro
df3['match_num'] = df3['match_num'].astype(int)

# ordena o dataframe por data e identificação do jogo
df3.sort_values(['tourney_date','match_num'],inplace=True)
# print(df3)

return df3

def obtem_dataframe_ODDS(ano:str) -> pd.DataFrame:

    # TRIM(jogador), "." na ultima posicao
    URL='http://www.tennis-data.co.uk/{}/{}.zip'.format(ano,ano)

    get_file(URL, '/Users/gilberto/Downloads/', 'ODDS_{}.zip'.format(ano))

    lst_files = ZipFile('/Users/gilberto/Downloads/ODDS_{}.zip'.format(ano)).namelist()

    unzip_file
    ('/Users/gilberto/Downloads/ODDS_{}.zip'.format(ano),'/Users/gilberto/Downloads/')

    df = pd.read_excel('/Users/gilberto/Downloads/' + lst_files[0], sheet_name=ano,
skiprows = 0, index_col=None)

    # adiciona a coluna Year (Ano)
    df['Year']=int(ano)

```

```

# =====
# Limpeza da base
# =====

# Retira caracteres em branco do nome do vencedor e do perdedor
# Adiciona um "." ao final de cada nome do vencedor e do perdedor
# substitui no nome do vencedor e do perdedor ".." por "." desta forma garante que
todos os nomes terminem com "."

df['Winner']= df['Winner'].str.strip()
df['Winner']= df['Winner'].map(str) + '.'
df['Winner'].replace('\.\.', '.', inplace=True, regex=True)

df['Loser']= df['Loser'].str.strip()
df['Loser']= df['Loser'].map(str) + '.'
df['Loser'].replace('\.\.', '.', inplace=True, regex=True)

if ano < '2010':
    insere_colunas_odds_medios(df)

#
=====
=====
# Insere colunas de identificação de torneios e jogadores para permitir
futuramente a junção
# com a base de resultados da ATP
#
=====
=====

#
df.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\ODDS_XXX
_2001.csv', index = False)

```

```
# dfx = df.loc[df['Winner'].isin(['Bahrouzian O.','Gromley C.','Hantschek M.','Kutac
R.','Luncanu P.A.','Marin L.','Stepanek M.'])]
# dfy = df.loc[df['Loser'].isin(['Bahrouzian O.','Gromley C.','Hantschek M.','Kutac
R.','Luncanu P.A.','Marin L.','Stepanek M.'])]
```

```
# dfx=pd.concat([dfx,dfy])
```

```
# print(dfx)
```

```
#
```

```
dfx.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\ODDS_ZZ
ZZ_' + str(ano) + '.csv', index = False)
```

```
return df
```

```
def gera_lista_jogadores_arquivo_ODDS_em_EXCEL():
```

```
    set_players = set()
```

```
    for ano in range(2001,2020):
```

```
        print(ano)
```

```
        df_odds = obtem_dataframe_ODDS(str(ano))
```

```
        w_set_players = set(df_odds.Winner)
```

```
        l_set_players = set(df_odds.Loser)
```

```
        set_players=set_players.union(w_set_players).union(l_set_players)
```

```
    df_players = pd.DataFrame(sorted(set_players))
```

```
    df_players.to_excel("C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\pl
ayers_odds.xlsx")
```

```
def gera_lista_jogadores_arquivo_results_em_EXCEL():
```

```

df_players =
pd.DataFrame(columns=['player_id','player_name','player_abrev_name'])
for ano in range(2001,2020):
    print(ano)

    df_results = obtem_dataframe_resultados(str(ano))

    df_winners_year = df_results[['winner_id', 'winner_name']].copy()
    df_winners_year.rename(columns={'winner_id': 'player_id', 'winner_name':
'player_name'}, inplace=True)

    df_players = pd.concat([df_winners_year,
df_players],ignore_index=True).drop_duplicates().reset_index(drop=True)

    df_losers_year = df_results[['loser_id', 'loser_name']].copy()
    df_losers_year.rename(columns={'loser_id': 'player_id', 'loser_name':
'player_name'}, inplace=True)

    df_players = pd.concat([df_losers_year,
df_players],ignore_index=True).drop_duplicates().reset_index(drop=True)

    # print(obtem_nome_abreviado(list(df_players['player_name'])))
    df_players['player_abrev_name'] =
obtem_nome_abreviado(list(df_players['player_name']))

df_players.to_excel("C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\pl
ayers_results.xlsx")

print('OK')

def gera_lista_torneios_arquivo_ODDS_em_EXCEL():

    for ano in range(2001,2020):

```

```

print(ano)

df_odds = obtem_dataframe_ODDS(str(ano))
df_tourneys = df_odds[['Year', 'ATP', 'Location', 'Tournament']].copy()
df_tourneys.drop_duplicates(keep = 'first', inplace = True)

if ano == 2001:
    df_all_tourneys = df_tourneys
else:
    df_all_tourneys =
pd.concat([df_all_tourneys,df_tourneys],ignore_index=True).drop_duplicates().reset_index(drop=True)

# print(df_all_tourneys)

df_all_tourneys.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\tourneys_odds.xlsx")

def gera_lista_torneios_arquivo_results_em_EXCEL():

    for ano in range(2001,2020):
        print(ano)

        df_results = obtem_dataframe_resultados(str(ano))
        df_tourneys = df_results[['Year', 'tourney_id', 'tourney_name']].copy()
        df_tourneys.drop_duplicates(keep = 'first', inplace = True)

        if ano == 2001:
            df_all_tourneys = df_tourneys
        else:
            df_all_tourneys =
pd.concat([df_all_tourneys,df_tourneys],ignore_index=True).drop_duplicates().reset_index(drop=True)

```

```

#print(df_all_tourneys)

df_all_tourneys.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\tourneys_results.xlsx")

def union_df_unknown_players():
    for ano in range(2001,2020):
        df = pd.read_csv("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\ODD
S_ZZZZ_" + str(ano) + ".csv", sep=',', delimiter=None, header='infer', names=None,
index_col=None)
        dfx = df[['Year', 'Location', 'Tournament', 'Date', 'Winner', 'Loser',
'W1','L1','W2','L2','W3','L3','W4','L4','W5', 'L5']].copy()
        if ano == 2001:
            dfy = dfx
        else:
            dfy=pd.concat([dfy,dfx])
        print(dfy)

dfy.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\crazy
_players.xlsx")

def merge_atp_and_odds_info_old():

    warnings.filterwarnings("ignore")

    # Obtem os dataframes de de=para
    # -----

    df_de_para_players = pd.read_excel('C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\TCC\\Meu_

```



```

Trabalho\players_de_para.xlsx',    sheet_name='DE_PARA',    skiprows    =    0,
index_col=None)

df_de_para_tourneys                                =
pd.read_excel('C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\TCC\\Meu_
Trabalho\\tourneys_de_para.xlsx',    sheet_name='DE_PARA',    skiprows    =    0,
index_col=None)

df_de_para_round                                =
pd.read_excel('C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\TCC\\Meu_
Trabalho\\round_de_para.xlsx',    sheet_name='DE_PARA',    skiprows    =    0,
index_col=None)

```

```

for ano in range(2000,2020):

```

```

    print(' ')
    print(str(ano))

```

```

    # Obtem o winner_id
    # -----

```

```

    df_odds = obtem_dataframe_ODDS(str(ano))

```

```

    df_1    =    df_odds.merge(df_de_para_players,    how='left',    left_on=['Winner'],
right_on=['Player_Name'])

```

```

    df_NaN = df_1[df_1['Player_ID'].isnull()]

```

```

    if not df_NaN.empty:
        print('Lista de winners contém nulos')
        print(df_NaN['Winner'] + ' ' + df_NaN['Loser'])
        break

```

```

    df_1.rename(columns={'Player_ID': 'winner_id'}, inplace=True)

```

```

    # Obtem o loser_id

```

```

# -----

df_2 = df_1.merge(df_de_para_players, how='left', left_on=['Loser'],
right_on=['Player_Name'])

df_NaN = df_2[df_2['Player_ID'].isnull()]

if not df_NaN.empty:
    print('Lista de losers contém nulos')
    print(df_NaN['Winner'] + ' ' + df_NaN['Loser'])
    break

df_2.rename(columns={'Player_ID': 'loser_id'}, inplace=True)

# Merge com de-para de torneios
# -----

df_3 = df_2.merge(df_de_para_tourneys, how='left', left_on=['Year', 'ATP',
'Location','Tournament'], right_on=['Year', 'ATP', 'Location','Tournament'])

df_3['tourney_id'] = df_3['tourney_id'].astype("string")
df_3['Round'] = df_3['Round'].astype("string")

# dataTypeSeries = df_3.dtypes
# print('Data type of each column of Dataframe :')
# print(dataTypeSeries)

df_NaN = df_3[df_3['tourney_id'].isnull()]

if not df_NaN.empty:
    print('Lista de tourneys contém nulos')
    print(df_NaN)
    break

```

```

# Obtem dataframe de resultados ATP
# -----

df_results = obtem_dataframe_resultados(str(ano))

df_results['round'] = df_results['round'].astype("string")
df_results['draw_size'] = df_results['draw_size'].astype(int)

df_4 = df_results.merge(df_de_para_round, how='left', left_on=['draw_size',
'round'], right_on=['draw_size', 'round'])

df_NaN = df_4[df_4['round_odds'].isnull()]

if not df_NaN.empty:
    print('Lista de rounds contém nulos')
    print(df_NaN)
    break

df_4['tourney_id'] = df_4['tourney_id'].astype("string")
df_4['winner_id'] = df_4['winner_id'].astype(int)
df_4['loser_id'] = df_4['loser_id'].astype(int)
df_4['round_odds'] = df_4['round_odds'].astype("string")

# dataTypeSeries = df_4.dtypes
# print('Data type of each column of Dataframe :')
# print(dataTypeSeries)

# df_merged_3_4 = df_3.merge(df_4, how='left', left_on=['Year', 'tourney_id',
'Round', 'winner_id', 'loser_id'], right_on=['Year', 'tourney_id', 'round_odds',
'winner_id', 'loser_id'])

# df_NaN = df_merged_3_4[df_merged_3_4['draw_size'].isnull()]

# if not df_NaN.empty:

```

```

# print('Lista de resultados contém nulos')
# print(df_NaN)
#
df_NaN.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\
merged_3_4_NAN.xlsx")

df_merged_4_3 = df_4.merge(df_3, how='left', right_on=['Year', 'tournament_id',
'Round', 'winner_id', 'loser_id'], left_on=['Year', 'tournament_id', 'round_odds',
'winner_id', 'loser_id'])

df_merged_4_3.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\
Python\\merged_ATP_ODDS.xlsx")

print('OK')

# df_NaN = df_merged_4_3[df_merged_4_3['W1'].isnull()]

# if not df_NaN.empty:
# print('Lista de odds contém nulos')
# print(df_NaN)
#
df_NaN.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\
merged_4_3_NAN.xlsx")

# print(df_merged)

def merge_atp_and_odds_info():

warnings.filterwarnings("ignore")

df_merged = pd.DataFrame

```

```

# Obtem os dataframes de de=para
# -----

df_de_para_players =
pd.read_excel('C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\TCC\\Meu_
Trabalho\\players_de_para.xlsx', sheet_name='DE_PARA', skiprows = 0,
index_col=None)

df_de_para_tourneys =
pd.read_excel('C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\TCC\\Meu_
Trabalho\\tourneys_de_para.xlsx', sheet_name='DE_PARA', skiprows = 0,
index_col=None)

df_de_para_round =
pd.read_excel('C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\TCC\\Meu_
Trabalho\\round_de_para.xlsx', sheet_name='DE_PARA', skiprows = 0,
index_col=None)

for ano in range(2000,2020):

    # print(' ')
    print(str(ano))

    df_odds = obtem_dataframe_ODDS(str(ano))

    # Obtem o winner_id
    # -----

    df_1 = df_odds.merge(df_de_para_players, how='left', left_on=['Winner'],
right_on=['Player_Name'])

    df_1.rename(columns={'Player_ID': 'winner_id'}, inplace=True)

    # Obtem o loser_id
    # -----

```

```
df_2 = df_1.merge(df_de_para_players, how='left', left_on=['Loser'],
right_on=['Player_Name'])
```

```
df_2.rename(columns={'Player_ID': 'loser_id'}, inplace=True)
```

```
# Merge com de-para de torneios
```

```
# -----
```

```
df_3 = df_2.merge(df_de_para_tourneys, how='left', left_on=['Year', 'ATP',
'Location','Tournament'], right_on=['Year', 'ATP', 'Location','Tournament'])
```

```
df_3['tourney_id'] = df_3['tourney_id'].astype("string")
```

```
df_3['Round'] = df_3['Round'].astype("string")
```

```
# Obtem dataframe de resultados ATP
```

```
# -----
```

```
df_results = obter_dataframe_resultados(str(ano))
```

```
df_results['round'] = df_results['round'].astype("string")
```

```
df_results['draw_size'] = df_results['draw_size'].astype(int)
```

```
df_4 = df_results.merge(df_de_para_round, how='left', left_on=['draw_size',
'round'], right_on=['draw_size', 'round'])
```

```
df_4['tourney_id'] = df_4['tourney_id'].astype("string")
```

```
df_4['winner_id'] = df_4['winner_id'].astype(int)
```

```
df_4['loser_id'] = df_4['loser_id'].astype(int)
```

```
df_4['round_odds'] = df_4['round_odds'].astype("string")
```

```
df_merged_4_3 = df_4.merge(df_3, how='left', right_on=['Year', 'tourney_id',
'Round', 'winner_id','loser_id'], left_on=['Year', 'tourney_id', 'round_odds',
'winner_id','loser_id'])
```

```

if not df_merged.empty:
    df_merged = df_merged.append(df_merged_4_3)
else:
    df_merged = df_merged_4_3

# Retira os jogos da Taça Davis pois para esses jogos não há apostas
# -----

df_merged = df_merged[df_merged['tourney_level'] != 'D']

df_merged.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\merged_ATP_ODDS.xlsx")

print('OK')

def gera_dataframes_resultados_agregados():

    warnings.filterwarnings("ignore")

    # cria dataframe de dados agregados por jogador

    columns_1 = {
        'player_id':",
        'tourney_id':",
        'tourney_date':",
        'round':",
        'match_num':int,
        'match_seq':int,
        'qty_grass_matches':int,
        'qty_grass_victories':int,
        'pct_grass_victories':float,
        'qty_hard_matches':int,

```

'qty\_hard\_victories':int,  
'pct\_hard\_victories':float,  
'qty\_clay\_matches':int,  
'qty\_clay\_victories':int,  
'pct\_clay\_victories':float,  
'qty\_matches\_against\_left\_handed':int,  
'qty\_victories\_against\_left\_handed':int,  
'pct\_victories\_against\_left\_handed':float,  
'qty\_3sets\_matches':int,  
'qty\_3sets\_victories':int,  
'pct\_3sets\_victories':float,  
'qty\_5sets\_matches':int,  
'qty\_5sets\_victories':int,  
'pct\_5sets\_victories':float,  
'minutes':int,  
'avg\_minutes':float,  
'qty\_ace':int,  
'qty\_df':int,  
'qty\_svpt':int,  
'qty\_1stIn':int,  
'qty\_1stWon':int,  
'qty\_2ndWon':int,  
'qty\_SvGms':int,  
'qty\_bpSaved':int,  
'qty\_bpFaced':int,  
'avg\_ace':float,  
'avg\_df':float,  
'avg\_svpt':float,  
'avg\_1stIn':float,  
'avg\_1stWon':float,  
'avg\_2ndWon':float,  
'avg\_SvGms':float,  
'avg\_bpSaved':float,  
'avg\_bpFaced':float,



```

'height':int,
'age':float,
'rank_points':int,
'qty_matches_grand_slam':int,
'qty_victories_grand_slam':int,
'pct_victories_grand_slam':float,
'qty_matches_master_1000':int,
'qty_victories_master_1000':int,
'pct_victories_master_1000':float,
'qty_matches_ATP_250_500':int,
'qty_victories_ATP_250_500':int,
'pct_victories_ATP_250_500':float,
'qty_matches_finals':int,
'qty_victories_finals':int,
'pct_victories_finals':float,
'qty_grand_slam_titles':int,
'qty_master_1000_titles':int,
'qty_ATP_250_500_titles':int,
'qty_finals_titles':int
}

```

```
df_1_player = pd.DataFrame(columns_1, index=[])
```

```
df_1_player_geral = pd.DataFrame(columns_1, index=[])
```

```
# cria dataframe de dados agregados por par de jogadores
```

```

columns_2 = {
    'player_1_id':",
    'player_2_id':",
    'tourney_id':",
    'tourney_date':",
    'round':",
    'match_num':int,
    'match_seq':int,

```

```

'qty_grass_matches':int,
'qty_player_1_grass_victories':int,
'pct_player_1_grass_victories':float,
'qty_hard_matches':int,
'qty_player_1_hard_victories':int,
'pct_player_1_hard_victories':float,
'qty_clay_matches':int,
'qty_player_1_clay_victories':int,
'pct_player_1_clay_victories':float,
'qty_3sets_matches':int,
'qty_player_1_3sets_victories':int,
'pct_player_1_3sets_victories':float,
'qty_5sets_matches':int,
'qty_player_1_5sets_victories':int,
'pct_player_1_5sets_victories':float,
'qty_grand_slam_matches':int,
'qty_player_1_grand_slam_victories':int,
'pct_player_1_grand_slam_victories':float,
'qty_master_1000_matches':int,
'qty_player_1_master_1000_victories':int,
'pct_player_1_master_1000_victories':float,
'qty_finals_matches':int,
'qty_player_1_finals_victories':int,
'pct_player_1_finals_victories':float,
'qty_ATP_250_500_matches':int,
'qty_player_1_ATP_250_500_victories':int,
'pct_player_1_ATP_250_500_victories':float
}

```

```

df_2_player = pd.DataFrame(columns_2, index=[])
df_2_player_geral = pd.DataFrame(columns_2, index=[])

```

```

for ano in range(1998,2020):

```

```

df_results = obter_dataframe_resultados(str(ano))

print('-----')
print(str(ano))
now = datetime.now()
print(now.strftime("%H:%M:%S"))
print(str(len(df_results.index)))
print('-----')
contador = 0

for ind in df_results.index:

    contador = contador + 1

    if (contador % 100) == 1:
        now = datetime.now()
        print(str(contador) + '-' + now.strftime("%H:%M:%S"))

    #if contador == 101:
    #    break

    # df3x = df3[(df3['winner_id'] == '103720') & (df3['match_num'] == 1)]

    # ##### #
    #      Gera dados agregados do vencedor      #
    # ##### #

    df_1_player_aux_1 = df_1_player[df_1_player['player_id'] ==
df_results.loc[ind,'winner_id']]

    if not df_1_player_aux_1.empty:

df_1_player_aux_1.sort_values(['match_seq'],inplace=True,ascending=[False])
df_1_player_aux_1.reset_index(drop=True,inplace=True)

```

```

# df_1_player_aux_1.reset_index(inplace=True)

#         print('Winner         encontrado         *****         :{)
'.format(df_results.loc[ind,'winner_id']))

df_1_player_aux_1.loc[0,"tourney_id"] = df_results.loc[ind,'tourney_id']
df_1_player_aux_1.loc[0,"tourney_date"] = df_results.loc[ind,'tourney_date']
df_1_player_aux_1.loc[0,"match_num"] = df_results.loc[ind,'match_num']
df_1_player_aux_1.loc[0,"round"] = df_results.loc[ind,'round']

else:
    # print('Winner não encontrado :{)'.format(df_results.loc[ind,'winner_id']))

df_1_player_aux_1 = pd.DataFrame([[
    df_results.loc[ind,'winner_id'],
    df_results.loc[ind,'tourney_id'],
    df_results.loc[ind,'tourney_date'],
    df_results.loc[ind,'round'],
    df_results.loc[ind,'match_num'],
    0 , # match_seq
    0 , # qty_grass_matches
    0 , # qty_grass_victories
    0.0 , # pct_grass_victories
    0 , # qty_hard_matches
    0 , # qty_hard_victories
    0.0 , # pct_hard_victories
    0 , # qty_clay_matches
    0 , # qty_clay_victories
    0.0 , # pct_clay_victories
    0 , # qty_matches_against_left_handed
    0 , # qty_victories_against_left_handed
    0.0 , # pct_victories_against_left_handed
    0 , # qty_3sets_matches
    0 , # qty_3sets_victories

```

```

0.0 , # pct_3sets_victories
0 , # qty_5sets_matches
0 , # qty_5sets_victories
0.0 , # pct_5sets_victories
0 , # minutes
0.0 , # avg_minutes
0 , # qty_ace
0 , # qty_df
0 , # qty_svpt
0 , # qty_1stIn
0 , # qty_1stWon
0 , # qty_2ndWon
0 , # qty_SvGms
0 , # qty_bpSaved
0 , # qty_bpFaced
0.0 , # avg_ace
0.0 , # avg_df
0.0 , # avg_svpt
0.0 , # avg_1stIn
0.0 , # avg_1stWon
0.0 , # avg_2ndWon
0.0 , # avg_SvGms
0.0 , # avg_bpSaved
0.0 , # avg_bpFaced
0.0 , # height
0.0 , # age
0 , # rank_points
0 , # qty_matches_grand_slam
0 , # qty_victories_grand_slam
0.0 , # pct_victories_grand_slam
0 , # qty_matches_master_1000
0 , # qty_victories_master_1000
0.0 , # pct_victories_master_1000
0 , # qty_matches_ATP_250_500

```

```

0 , # qty_victories_ATP_250_500
0.0 , # pct_victories_ATP_250_500
0 , # qty_matches_finals
0 , # qty_victories_finals
0.0 , # pct_victories_finals
0 , # qty_grand_slam_titles
0 , # qty_master_1000_titles
0 , # qty_ATP_250_500_titles
0 # qty_finals_titles
]], columns = columns_1 )

```

```

df_1_player_aux_1.loc[0,"match_seq"] =
df_1_player_aux_1.loc[0,"match_seq"] + 1

```

```

if df_results.loc[ind,'surface'] == 'Clay':
    df_1_player_aux_1.loc[0,"qty_clay_matches"] =
df_1_player_aux_1.loc[0,"qty_clay_matches"] + 1
    df_1_player_aux_1.loc[0,"qty_clay_victories"] =
df_1_player_aux_1.loc[0,"qty_clay_victories"] + 1
    df_1_player_aux_1.loc[0,"pct_clay_victories"] =
df_1_player_aux_1.loc[0,"qty_clay_victories"] /
df_1_player_aux_1.loc[0,"qty_clay_matches"] * 100
elif df_results.loc[ind,'surface'] == 'Grass':
    df_1_player_aux_1.loc[0,"qty_grass_matches"] =
df_1_player_aux_1.loc[0,"qty_grass_matches"] + 1
    df_1_player_aux_1.loc[0,"qty_grass_victories"] =
df_1_player_aux_1.loc[0,"qty_grass_victories"] + 1
    df_1_player_aux_1.loc[0,"pct_grass_victories"] =
df_1_player_aux_1.loc[0,"qty_grass_victories"] /
df_1_player_aux_1.loc[0,"qty_grass_matches"] * 100
else:
    df_1_player_aux_1.loc[0,"qty_hard_matches"] =
df_1_player_aux_1.loc[0,"qty_hard_matches"] + 1

```

```

df_1_player_aux_1.loc[0,"qty_hard_victories"] =
df_1_player_aux_1.loc[0,"qty_hard_victories"] + 1
df_1_player_aux_1.loc[0,"pct_hard_victories"] =
df_1_player_aux_1.loc[0,"qty_hard_victories"] /
df_1_player_aux_1.loc[0,"qty_hard_matches"] * 100

```

```

if df_results.loc[ind,'loser_hand'] == 'L':
    df_1_player_aux_1.loc[0,"qty_matches_against_left_handed"] =
df_1_player_aux_1.loc[0,"qty_matches_against_left_handed"] + 1
    df_1_player_aux_1.loc[0,"qty_victories_against_left_handed"] =
df_1_player_aux_1.loc[0,"qty_victories_against_left_handed"] + 1
    df_1_player_aux_1.loc[0,"pct_victories_against_left_handed"] =
df_1_player_aux_1.loc[0,"qty_victories_against_left_handed"] /
df_1_player_aux_1.loc[0,"qty_matches_against_left_handed"] * 100

```

```

if df_results.loc[ind,'best_of'] == '3':
    df_1_player_aux_1.loc[0,"qty_3sets_matches"] =
df_1_player_aux_1.loc[0,"qty_3sets_matches"] + 1
    df_1_player_aux_1.loc[0,"qty_3sets_victories"] =
df_1_player_aux_1.loc[0,"qty_3sets_victories"] + 1
    df_1_player_aux_1.loc[0,"pct_3sets_victories"] =
df_1_player_aux_1.loc[0,"qty_3sets_victories"] /
df_1_player_aux_1.loc[0,"qty_3sets_matches"] * 100
else:
    df_1_player_aux_1.loc[0,"qty_5sets_matches"] =
df_1_player_aux_1.loc[0,"qty_5sets_matches"] + 1
    df_1_player_aux_1.loc[0,"qty_5sets_victories"] =
df_1_player_aux_1.loc[0,"qty_5sets_victories"] + 1
    df_1_player_aux_1.loc[0,"pct_5sets_victories"] =
df_1_player_aux_1.loc[0,"qty_5sets_victories"] /
df_1_player_aux_1.loc[0,"qty_5sets_matches"] * 100

```

```
df_1_player_aux_1.loc[0,"minutes"] = df_1_player_aux_1.loc[0,"minutes"] +
int("0" + df_results.loc[ind,'minutes'])
```

```
df_1_player_aux_1.loc[0,"avg_minutes"] =
df_1_player_aux_1.loc[0,"minutes"] / df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"qty_ace"] = df_1_player_aux_1.loc[0,"qty_ace"] +
int("0" + df_results.loc[ind,'w_ace'])
```

```
df_1_player_aux_1.loc[0,"qty_df"] = df_1_player_aux_1.loc[0,"qty_df"] +
int("0" + df_results.loc[ind,'w_df'])
```

```
df_1_player_aux_1.loc[0,"qty_svpt"] = df_1_player_aux_1.loc[0,"qty_svpt"] +
int("0" + df_results.loc[ind,'w_svpt'])
```

```
df_1_player_aux_1.loc[0,"qty_1stIn"] = df_1_player_aux_1.loc[0,"qty_1stIn"] +
int("0" + df_results.loc[ind,'w_1stIn'])
```

```
df_1_player_aux_1.loc[0,"qty_1stWon"] =
df_1_player_aux_1.loc[0,"qty_1stWon"] + int("0" + df_results.loc[ind,'w_1stWon'])
```

```
df_1_player_aux_1.loc[0,"qty_2ndWon"] =
df_1_player_aux_1.loc[0,"qty_2ndWon"] + int("0" + df_results.loc[ind,'w_2ndWon'])
```

```
df_1_player_aux_1.loc[0,"qty_SvGms"] =
df_1_player_aux_1.loc[0,"qty_SvGms"] + int("0" + df_results.loc[ind,'w_SvGms'])
```

```
df_1_player_aux_1.loc[0,"qty_bpSaved"] =
df_1_player_aux_1.loc[0,"qty_bpSaved"] + int("0" + df_results.loc[ind,'w_bpSaved'])
```

```
df_1_player_aux_1.loc[0,"qty_bpFaced"] =
df_1_player_aux_1.loc[0,"qty_bpFaced"] + int("0" + df_results.loc[ind,'w_bpFaced'])
```

```
df_1_player_aux_1.loc[0,"avg_ace"] = df_1_player_aux_1.loc[0,"qty_ace"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_df"] = df_1_player_aux_1.loc[0,"qty_df"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_svpt"] = df_1_player_aux_1.loc[0,"qty_svpt"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_1stIn"] = df_1_player_aux_1.loc[0,"qty_1stIn"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_1stWon"] =
df_1_player_aux_1.loc[0,"qty_1stWon"] / df_1_player_aux_1.loc[0,"match_seq"]
```



```

df_1_player_aux_1.loc[0,"avg_2ndWon"] =
df_1_player_aux_1.loc[0,"qty_2ndWon"] / df_1_player_aux_1.loc[0,"match_seq"]
df_1_player_aux_1.loc[0,"avg_SvGms"] =
df_1_player_aux_1.loc[0,"qty_SvGms"] / df_1_player_aux_1.loc[0,"match_seq"]
df_1_player_aux_1.loc[0,"avg_bpSaved"] =
df_1_player_aux_1.loc[0,"qty_bpSaved"] / df_1_player_aux_1.loc[0,"match_seq"]
df_1_player_aux_1.loc[0,"avg_bpFaced"] =
df_1_player_aux_1.loc[0,"qty_bpFaced"] / df_1_player_aux_1.loc[0,"match_seq"]

df_1_player_aux_1.loc[0,"height"] = int("0" + df_results.loc[ind,'winner_ht'])
df_1_player_aux_1.loc[0,"age"] = float("0" + df_results.loc[ind,'winner_age'])
df_1_player_aux_1.loc[0,"rank_points"] = int("0" +
df_results.loc[ind,'winner_rank_points'])

if df_results.loc[ind,'tourney_level'] == 'A': # ATP 250, 500
df_1_player_aux_1.loc[0,"qty_matches_ATP_250_500"] =
df_1_player_aux_1.loc[0,"qty_matches_ATP_250_500"] + 1
df_1_player_aux_1.loc[0,"qty_victories_ATP_250_500"] =
df_1_player_aux_1.loc[0,"qty_victories_ATP_250_500"] + 1
df_1_player_aux_1.loc[0,"pct_victories_ATP_250_500"] =
df_1_player_aux_1.loc[0,"qty_victories_ATP_250_500"] /
df_1_player_aux_1.loc[0,"qty_matches_ATP_250_500"] * 100
if df_results.loc[ind,'round'] == 'F': # Partida final do torneio
df_1_player_aux_1.loc[0,"qty_ATP_250_500_titles"] =
df_1_player_aux_1.loc[0,"qty_ATP_250_500_titles"] + 1

elif df_results.loc[ind,'tourney_level'] == 'F': # ATP Finals
df_1_player_aux_1.loc[0,"qty_matches_finals"] =
df_1_player_aux_1.loc[0,"qty_matches_finals"] + 1
df_1_player_aux_1.loc[0,"qty_victories_finals"] =
df_1_player_aux_1.loc[0,"qty_victories_finals"] + 1
df_1_player_aux_1.loc[0,"pct_victories_finals"] =
df_1_player_aux_1.loc[0,"qty_victories_finals"] /
df_1_player_aux_1.loc[0,"qty_matches_finals"] * 100

```

```

        if df_results.loc[ind,'round'] == 'F': # Partida final do torneio
            df_1_player_aux_1.loc[0,"qty_finals_titles"] =
df_1_player_aux_1.loc[0,"qty_finals_titles"] + 1

        elif df_results.loc[ind,'tourney_level'] == 'G': # Grand Slam
            df_1_player_aux_1.loc[0,"qty_matches_grand_slam"] =
df_1_player_aux_1.loc[0,"qty_matches_grand_slam"] + 1
            df_1_player_aux_1.loc[0,"qty_victories_grand_slam"] =
df_1_player_aux_1.loc[0,"qty_victories_grand_slam"] + 1
            df_1_player_aux_1.loc[0,"pct_victories_grand_slam"] =
df_1_player_aux_1.loc[0,"qty_victories_grand_slam"] /
df_1_player_aux_1.loc[0,"qty_matches_grand_slam"] * 100
            if df_results.loc[ind,'round'] == 'F': # Partida final do torneio
                df_1_player_aux_1.loc[0,"qty_grand_slam_titles"] =
df_1_player_aux_1.loc[0,"qty_grand_slam_titles"] + 1

        elif df_results.loc[ind,'tourney_level'] == 'M': # Master 1000
            df_1_player_aux_1.loc[0,"qty_matches_master_1000"] =
df_1_player_aux_1.loc[0,"qty_matches_master_1000"] + 1
            df_1_player_aux_1.loc[0,"qty_victories_master_1000"] =
df_1_player_aux_1.loc[0,"qty_victories_master_1000"] + 1
            df_1_player_aux_1.loc[0,"pct_victories_master_1000"] =
df_1_player_aux_1.loc[0,"qty_victories_master_1000"] /
df_1_player_aux_1.loc[0,"qty_matches_master_1000"] * 100
            if df_results.loc[ind,'round'] == 'F': # Partida final do torneio
                df_1_player_aux_1.loc[0,"qty_master_1000_titles"] =
df_1_player_aux_1.loc[0,"qty_master_1000_titles"] + 1

df_1_player = df_1_player.append(df_1_player_aux_1.iloc[0])
df_1_player_geral = df_1_player_geral.append(df_1_player_aux_1.iloc[0])
del df_1_player_aux_1

# ##### #
#      Gera dados agregados do perdedor      #

```

```

##### #

df_1_player_aux_1      =      df_1_player[df_1_player['player_id']      ==
df_results.loc[ind,'loser_id']]

if not df_1_player_aux_1.empty:

df_1_player_aux_1.sort_values(['match_seq'],inplace=True,ascending=[False])
df_1_player_aux_1.reset_index(drop=True,inplace=True)
# df_1_player_aux_1.reset_index(inplace=True)

# print('Loser encontrado ***** :{}'.format(df_results.loc[ind,'loser_id']))

# if ano == 2019:
#   print(df_1_player_aux_1)
#   print('Tourney : ' + df_results.loc[ind,'tourney_id'])
#   print('Torneio : ' + df_1_player_aux_1.loc[0,"tourney_id"])

df_1_player_aux_1.loc[0,"tourney_id"] = df_results.loc[ind,'tourney_id']
df_1_player_aux_1.loc[0,"tourney_date"] = df_results.loc[ind,'tourney_date']
df_1_player_aux_1.loc[0,"match_num"] = df_results.loc[ind,'match_num']
df_1_player_aux_1.loc[0,"round"] = df_results.loc[ind,'round']

else:
# print('Loser não encontrado :{}'.format(df_results.loc[ind,'loser_id']))

df_1_player_aux_1 = pd.DataFrame([[
df_results.loc[ind,'loser_id'],
df_results.loc[ind,'tourney_id'],
df_results.loc[ind,'tourney_date'],
df_results.loc[ind,'round'],
df_results.loc[ind,'match_num'],
0 , # match_seq

```

0 , # qty\_grass\_matches  
0 , # qty\_grass\_victories  
0.0 , # pct\_grass\_victories  
0 , # qty\_hard\_matches  
0 , # qty\_hard\_victories  
0.0 , # pct\_hard\_victories  
0 , # qty\_clay\_matches  
0 , # qty\_clay\_victories  
0.0 , # pct\_clay\_victories  
0 , # qty\_matches\_against\_left\_handed  
0 , # qty\_victories\_against\_left\_handed  
0.0 , # pct\_victories\_against\_left\_handed  
0 , # qty\_3sets\_matches  
0 , # qty\_3sets\_victories  
0.0 , # pct\_3sets\_victories  
0 , # qty\_5sets\_matches  
0 , # qty\_5sets\_victories  
0.0 , # pct\_5sets\_victories  
0 , # minutes  
0.0 , # avg\_minutes  
0 , # qty\_ace  
0 , # qty\_df  
0 , # qty\_svpt  
0 , # qty\_1stIn  
0 , # qty\_1stWon  
0 , # qty\_2ndWon  
0 , # qty\_SvGms  
0 , # qty\_bpSaved  
0 , # qty\_bpFaced  
0.0 , # avg\_ace  
0.0 , # avg\_df  
0.0 , # avg\_svpt  
0.0 , # avg\_1stIn  
0.0 , # avg\_1stWon

```

0.0 , # avg_2ndWon
0.0 , # avg_SvGms
0.0 , # avg_bpSaved
0.0 , # avg_bpFaced
0.0 , # height
0.0 , # age
0 , # rank_points
0 , # qty_matches_grand_slam
0 , # qty_victories_grand_slam
0.0 , # pct_victories_grand_slam
0 , # qty_matches_master_1000
0 , # qty_victories_master_1000
0.0 , # pct_victories_master_1000
0 , # qty_matches_ATP_250_500
0 , # qty_victories_ATP_250_500
0.0 , # pct_victories_ATP_250_500
0 , # qty_matches_finals
0 , # qty_victories_finals
0.0 , # pct_victories_finals
0 , # qty_grand_slam_titles
0 , # qty_master_1000_titles
0 , # qty_ATP_250_500_titles
0 , # qty_finals_titles
]], columns = columns_1 )

```

```

df_1_player_aux_1.loc[0,"match_seq"] =
df_1_player_aux_1.loc[0,"match_seq"] + 1

```

```

if df_results.loc[ind,'surface'] == 'Clay':
    df_1_player_aux_1.loc[0,"qty_clay_matches"] =
df_1_player_aux_1.loc[0,"qty_clay_matches"] + 1
    df_1_player_aux_1.loc[0,"pct_clay_victories"] =
df_1_player_aux_1.loc[0,"qty_clay_victories"] /
df_1_player_aux_1.loc[0,"qty_clay_matches"] * 100

```

```

elif df_results.loc[ind,'surface'] == 'Grass':
    df_1_player_aux_1.loc[0,"qty_grass_matches"] =
df_1_player_aux_1.loc[0,"qty_grass_matches"] + 1
    df_1_player_aux_1.loc[0,"pct_grass_victories"] =
df_1_player_aux_1.loc[0,"qty_grass_victories"] /
df_1_player_aux_1.loc[0,"qty_grass_matches"] * 100
else:
    df_1_player_aux_1.loc[0,"qty_hard_matches"] =
df_1_player_aux_1.loc[0,"qty_hard_matches"] + 1
    df_1_player_aux_1.loc[0,"pct_hard_victories"] =
df_1_player_aux_1.loc[0,"qty_hard_victories"] /
df_1_player_aux_1.loc[0,"qty_hard_matches"] * 100

if df_results.loc[ind,'winner_hand'] == 'L':
    df_1_player_aux_1.loc[0,"qty_matches_against_left_handed"] =
df_1_player_aux_1.loc[0,"qty_matches_against_left_handed"] + 1
    df_1_player_aux_1.loc[0,"pct_victories_against_left_handed"] =
df_1_player_aux_1.loc[0,"qty_victories_against_left_handed"] /
df_1_player_aux_1.loc[0,"qty_matches_against_left_handed"] * 100

if df_results.loc[ind,'best_of'] == '3':
    df_1_player_aux_1.loc[0,"qty_3sets_matches"] =
df_1_player_aux_1.loc[0,"qty_3sets_matches"] + 1
    df_1_player_aux_1.loc[0,"pct_3sets_victories"] =
df_1_player_aux_1.loc[0,"qty_3sets_victories"] /
df_1_player_aux_1.loc[0,"qty_3sets_matches"] * 100
else:
    df_1_player_aux_1.loc[0,"qty_5sets_matches"] =
df_1_player_aux_1.loc[0,"qty_5sets_matches"] + 1
    df_1_player_aux_1.loc[0,"pct_5sets_victories"] =
df_1_player_aux_1.loc[0,"qty_5sets_victories"] /
df_1_player_aux_1.loc[0,"qty_5sets_matches"] * 100

```

```
df_1_player_aux_1.loc[0,"minutes"] = df_1_player_aux_1.loc[0,"minutes"] +
int("0" + df_results.loc[ind,'minutes'])
```

```
df_1_player_aux_1.loc[0,"avg_minutes"] =
df_1_player_aux_1.loc[0,"minutes"] / df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"qty_ace"] = df_1_player_aux_1.loc[0,"qty_ace"] +
int("0" + df_results.loc[ind,'l_ace'])
```

```
df_1_player_aux_1.loc[0,"qty_df"] = df_1_player_aux_1.loc[0,"qty_df"] +
int("0" + df_results.loc[ind,'l_df'])
```

```
df_1_player_aux_1.loc[0,"qty_svpt"] = df_1_player_aux_1.loc[0,"qty_svpt"] +
int("0" + df_results.loc[ind,'l_svpt'])
```

```
df_1_player_aux_1.loc[0,"qty_1stIn"] = df_1_player_aux_1.loc[0,"qty_1stIn"] +
int("0" + df_results.loc[ind,'l_1stIn'])
```

```
df_1_player_aux_1.loc[0,"qty_1stWon"] =
df_1_player_aux_1.loc[0,"qty_1stWon"] + int("0" + df_results.loc[ind,'l_1stWon'])
```

```
df_1_player_aux_1.loc[0,"qty_2ndWon"] =
df_1_player_aux_1.loc[0,"qty_2ndWon"] + int("0" + df_results.loc[ind,'l_2ndWon'])
```

```
df_1_player_aux_1.loc[0,"qty_SvGms"] =
df_1_player_aux_1.loc[0,"qty_SvGms"] + int("0" + df_results.loc[ind,'l_SvGms'])
```

```
df_1_player_aux_1.loc[0,"qty_bpSaved"] =
df_1_player_aux_1.loc[0,"qty_bpSaved"] + int("0" + df_results.loc[ind,'l_bpSaved'])
```

```
df_1_player_aux_1.loc[0,"qty_bpFaced"] =
df_1_player_aux_1.loc[0,"qty_bpFaced"] + int("0" + df_results.loc[ind,'l_bpFaced'])
```

```
df_1_player_aux_1.loc[0,"avg_ace"] = df_1_player_aux_1.loc[0,"qty_ace"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_df"] = df_1_player_aux_1.loc[0,"qty_df"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_svpt"] = df_1_player_aux_1.loc[0,"qty_svpt"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_1stIn"] = df_1_player_aux_1.loc[0,"qty_1stIn"] /
df_1_player_aux_1.loc[0,"match_seq"]
```

```
df_1_player_aux_1.loc[0,"avg_1stWon"] =
df_1_player_aux_1.loc[0,"qty_1stWon"] / df_1_player_aux_1.loc[0,"match_seq"]
```

```

df_1_player_aux_1.loc[0,"avg_2ndWon"] =
df_1_player_aux_1.loc[0,"qty_2ndWon"] / df_1_player_aux_1.loc[0,"match_seq"]
df_1_player_aux_1.loc[0,"avg_SvGms"] =
df_1_player_aux_1.loc[0,"qty_SvGms"] / df_1_player_aux_1.loc[0,"match_seq"]
df_1_player_aux_1.loc[0,"avg_bpSaved"] =
df_1_player_aux_1.loc[0,"qty_bpSaved"] / df_1_player_aux_1.loc[0,"match_seq"]
df_1_player_aux_1.loc[0,"avg_bpFaced"] =
df_1_player_aux_1.loc[0,"qty_bpFaced"] / df_1_player_aux_1.loc[0,"match_seq"]

```

```

df_1_player_aux_1.loc[0,"height"] = int("0" + df_results.loc[ind,'loser_ht'])
df_1_player_aux_1.loc[0,"age"] = float("0" + df_results.loc[ind,'loser_age'])
df_1_player_aux_1.loc[0,"rank_points"] = int("0" +
df_results.loc[ind,'loser_rank_points'])

```

```

if df_results.loc[ind,'tourney_level'] == 'A': # ATP 250, 500
    df_1_player_aux_1.loc[0,"qty_matches_ATP_250_500"] =
df_1_player_aux_1.loc[0,"qty_matches_ATP_250_500"] + 1
    df_1_player_aux_1.loc[0,"pct_victories_ATP_250_500"] =
df_1_player_aux_1.loc[0,"qty_victories_ATP_250_500"] /
df_1_player_aux_1.loc[0,"qty_matches_ATP_250_500"] * 100
    elif df_results.loc[ind,'tourney_level'] == 'F': # ATP Finals
        df_1_player_aux_1.loc[0,"qty_matches_finals"] =
df_1_player_aux_1.loc[0,"qty_matches_finals"] + 1
        df_1_player_aux_1.loc[0,"pct_victories_finals"] =
df_1_player_aux_1.loc[0,"qty_victories_finals"] /
df_1_player_aux_1.loc[0,"qty_matches_finals"] * 100
        elif df_results.loc[ind,'tourney_level'] == 'G': # Grand Slam
            df_1_player_aux_1.loc[0,"qty_matches_grand_slam"] =
df_1_player_aux_1.loc[0,"qty_matches_grand_slam"] + 1
            df_1_player_aux_1.loc[0,"pct_victories_grand_slam"] =
df_1_player_aux_1.loc[0,"qty_victories_grand_slam"] /
df_1_player_aux_1.loc[0,"qty_matches_grand_slam"] * 100
            elif df_results.loc[ind,'tourney_level'] == 'M': # Master 1000

```



```

df_1_player_aux_1.loc[0,"qty_matches_master_1000"] =
df_1_player_aux_1.loc[0,"qty_matches_master_1000"] + 1
df_1_player_aux_1.loc[0,"pct_victories_master_1000"] =
df_1_player_aux_1.loc[0,"qty_victories_master_1000"] /
df_1_player_aux_1.loc[0,"qty_matches_master_1000"] * 100

```

```

df_1_player = df_1_player.append(df_1_player_aux_1.iloc[0])
df_1_player_geral = df_1_player_geral.append(df_1_player_aux_1.iloc[0])
del df_1_player_aux_1

```

```

#

```

```

##### #

```

```

#      Gera dados agregados do vencedor e perdedor juntos      #

```

```

#

```

```

##### #

```

```

if df_results.loc[ind,'winner_id'] < df_results.loc[ind,'loser_id']:

```

```

    player_1_id = df_results.loc[ind,'winner_id']

```

```

    player_2_id = df_results.loc[ind,'loser_id']

```

```

else:

```

```

    player_1_id = df_results.loc[ind,'loser_id']

```

```

    player_2_id = df_results.loc[ind,'winner_id']

```

```

# df3x = df3[(df3['winner_id'] == '103720') & (df3['match_num'] == 1)]

```

```

df_2_player_aux_1 = df_2_player[(df_2_player['player_1_id'] == player_1_id)
& (df_2_player['player_2_id'] == player_2_id)]

```

```

if not df_2_player_aux_1.empty:

```

```

df_2_player_aux_1.sort_values(['match_seq'],inplace=True,ascending=[False])

```

```

df_2_player_aux_1.reset_index(drop=True,inplace=True)

```

```

df_2_player_aux_1.loc[0,"tourney_id"] = df_results.loc[ind,'tourney_id']

```

```

df_2_player_aux_1.loc[0,"tourney_date"] = df_results.loc[ind,'tourney_date']
df_2_player_aux_1.loc[0,"match_num"] = df_results.loc[ind,'match_num']
df_2_player_aux_1.loc[0,"round"] = df_results.loc[ind,'round']

```

else:

```

df_2_player_aux_1 = pd.DataFrame([[
    player_1_id,
    player_2_id,
    df_results.loc[ind,'tourney_id'],
    df_results.loc[ind,'tourney_date'],
    df_results.loc[ind,'round'],
    df_results.loc[ind,'match_num'],
    0 , # match_seq
    0 , # qty_grass_matches
    0 , # qty_player_1_grass_victories
    0.0 , # pct_player_1_grass_victories
    0 , # qty_hard_matches
    0 , # qty_player_1_hard_victories
    0.0 , # pct_player_1_hard_victories
    0 , # qty_clay_matches
    0 , # qty_player_1_clay_victories
    0.0 , # pct_player_1_clay_victories
    0 , # qty_3sets_matches
    0 , # qty_player_1_3sets_victories
    0.0 , # pct_player_1_3sets_victories
    0 , # qty_5sets_matches
    0 , # qty_player_1_5sets_victories
    0.0 , # pct_player_1_5sets_victories
    0 , # qty_grand_slam_matches
    0 , # qty_player_1_grand_slam_victories
    0.0 , # pct_player_1_grand_slam_victories
    0 , # qty_master_1000_matches
    0 , # qty_player_1_master_1000_victories
    0.0 , # pct_player_1_master_1000_victories

```

```

0 , # qty_finals_matches
0 , # qty_player_1_finals_victories
0.0 , # pct_player_1_finals_victories
0 , # qty_ATP_250_500_matches
0 , # qty_player_1_ATP_250_500_victories
0.0 # pct_player_1_ATP_250_500_victories
]], columns = columns_2)

```

```

df_2_player_aux_1.loc[0,"match_seq"] =
df_2_player_aux_1.loc[0,"match_seq"] + 1

```

```

if df_results.loc[ind,'surface'] == 'Clay':
    df_2_player_aux_1.loc[0,"qty_clay_matches"] =
df_2_player_aux_1.loc[0,"qty_clay_matches"] + 1
    if player_1_id == df_results.loc[ind,'winner_id']:
        df_2_player_aux_1.loc[0,"qty_player_1_clay_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_clay_victories"] + 1
        df_2_player_aux_1.loc[0,"pct_player_1_clay_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_clay_victories"] /
df_2_player_aux_1.loc[0,"qty_clay_matches"] * 100
    elif df_results.loc[ind,'surface'] == 'Grass':
        df_2_player_aux_1.loc[0,"qty_grass_matches"] =
df_2_player_aux_1.loc[0,"qty_grass_matches"] + 1
        if player_1_id == df_results.loc[ind,'winner_id']:
            df_2_player_aux_1.loc[0,"qty_player_1_grass_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_grass_victories"] + 1
            df_2_player_aux_1.loc[0,"pct_player_1_grass_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_grass_victories"] /
df_2_player_aux_1.loc[0,"qty_grass_matches"] * 100
    else:
        df_2_player_aux_1.loc[0,"qty_hard_matches"] =
df_2_player_aux_1.loc[0,"qty_hard_matches"] + 1
        if player_1_id == df_results.loc[ind,'winner_id']:

```

```

df_2_player_aux_1.loc[0,"qty_player_1_hard_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_hard_victories"] + 1
df_2_player_aux_1.loc[0,"pct_player_1_hard_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_hard_victories"] /
df_2_player_aux_1.loc[0,"qty_hard_matches"] * 100

```

```

if df_results.loc[ind,'best_of'] == '3':
    df_2_player_aux_1.loc[0,"qty_3sets_matches"] =
df_2_player_aux_1.loc[0,"qty_3sets_matches"] + 1
    if player_1_id == df_results.loc[ind,'winner_id']:
        df_2_player_aux_1.loc[0,"qty_player_1_3sets_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_3sets_victories"] + 1
        df_2_player_aux_1.loc[0,"pct_player_1_3sets_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_3sets_victories"] /
df_2_player_aux_1.loc[0,"qty_3sets_matches"] * 100
    else:
        df_2_player_aux_1.loc[0,"qty_5sets_matches"] =
df_2_player_aux_1.loc[0,"qty_5sets_matches"] + 1
        if player_1_id == df_results.loc[ind,'winner_id']:
            df_2_player_aux_1.loc[0,"qty_player_1_5sets_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_5sets_victories"] + 1
            df_2_player_aux_1.loc[0,"pct_player_1_5sets_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_5sets_victories"] /
df_2_player_aux_1.loc[0,"qty_5sets_matches"] * 100

```

```

if df_results.loc[ind,'tourney_level'] == 'A': # ATP 250, 500
    df_2_player_aux_1.loc[0,"qty_ATP_250_500_matches"] =
df_2_player_aux_1.loc[0,"qty_ATP_250_500_matches"] + 1
    if player_1_id == df_results.loc[ind,'winner_id']:
        df_2_player_aux_1.loc[0,"qty_player_1_ATP_250_500_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_ATP_250_500_victories"] + 1

```

```

        df_2_player_aux_1.loc[0,"pct_player_1_ATP_250_500_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_ATP_250_500_victories"] /
df_2_player_aux_1.loc[0,"qty_ATP_250_500_matches"] * 100
        elif df_results.loc[ind,'tourney_level'] == 'F': # ATP Finals
            df_2_player_aux_1.loc[0,"qty_finals_matches"] =
df_2_player_aux_1.loc[0,"qty_finals_matches"] + 1
            if player_1_id == df_results.loc[ind,'winner_id']:
                df_2_player_aux_1.loc[0,"qty_player_1_finals_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_finals_victories"] + 1
                df_2_player_aux_1.loc[0,"pct_player_1_finals_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_finals_victories"] /
df_2_player_aux_1.loc[0,"qty_finals_matches"] * 100
            elif df_results.loc[ind,'tourney_level'] == 'G': # Grand Slam
                df_2_player_aux_1.loc[0,"qty_grand_slam_matches"] =
df_2_player_aux_1.loc[0,"qty_grand_slam_matches"] + 1
                if player_1_id == df_results.loc[ind,'winner_id']:
                    df_2_player_aux_1.loc[0,"qty_player_1_grand_slam_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_grand_slam_victories"] + 1
                    df_2_player_aux_1.loc[0,"pct_player_1_grand_slam_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_grand_slam_victories"] /
df_2_player_aux_1.loc[0,"qty_grand_slam_matches"] * 100
                elif df_results.loc[ind,'tourney_level'] == 'M': # Master 1000
                    df_2_player_aux_1.loc[0,"qty_master_1000_matches"] =
df_2_player_aux_1.loc[0,"qty_master_1000_matches"] + 1
                    if player_1_id == df_results.loc[ind,'winner_id']:
                        df_2_player_aux_1.loc[0,"qty_player_1_master_1000_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_master_1000_victories"] + 1
                        df_2_player_aux_1.loc[0,"pct_player_1_master_1000_victories"] =
df_2_player_aux_1.loc[0,"qty_player_1_master_1000_victories"] /
df_2_player_aux_1.loc[0,"qty_master_1000_matches"] * 100

df_2_player = df_2_player.append(df_2_player_aux_1.iloc[0])
df_2_player_geral = df_2_player_geral.append(df_2_player_aux_1.iloc[0])

```

```

del df_2_player_aux_1

df_1_player_last = df_1_player.groupby(['player_id'],
sort=False).agg({'match_seq':'max'})
df_1_player = df_1_player.merge(df_1_player_last, how='inner',
right_on=['player_id', 'match_seq'], left_on=['player_id', 'match_seq'])
# df_1_player.reset_index(inplace=True)
print(df_1_player)

df_1_player.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\df
1_player_' + str(ano) + '.csv', index = False)

df_2_player_last = df_2_player.groupby(['player_1_id', 'player_2_id'],
sort=False).agg({'match_seq':'max'})
df_2_player = df_2_player.merge(df_2_player_last, how='inner',
right_on=['player_1_id', 'player_2_id', 'match_seq'], left_on=['player_1_id',
'player_2_id', 'match_seq'])
# df_2_player.reset_index(inplace=True)

df_2_player.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\df
2_player_' + str(ano) + '.csv', index = False)

now = datetime.now()
print(now.strftime("%H:%M:%S"))

df_1_player_geral.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Pyt
hon\df1_player_geral' + '.csv', index = False)

df_2_player_geral.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Pyt
hon\df2_player_geral' + '.csv', index = False)
print("OK")

def select_winner (row):

```

```

if row['winner_id'] < row['loser_id'] :
    return 1
else:
    return 2

```

```

def select_winner_player_1_2 (row):
    if row['winner_id'] < row['loser_id'] :
        row['winner'] = 1
        row['player_1_id'] = row['winner_id']
        row['player_2_id'] = row['loser_id']
    else:
        row['winner'] = 2
        row['player_1_id'] = row['loser_id']
        row['player_2_id'] = row['winner_id']

    return row

```

```

def select_player_1 (row):
    if row['winner_id'] < row['loser_id'] :
        return row['winner_id']
    else:
        return row['loser_id']

```

```

def select_player_2 (row):
    if row['winner_id'] < row['loser_id'] :
        return row['loser_id']
    else:
        return row['winner_id']

```

```

def gera_ATP_ODDS_final():

```

```

# Obtem os dados de merge de resultados da ATP e odds de apostas
# -----

df_ATP_ODDS = pd.read_excel(
(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\Saved_2\merged_AT
P_ODDS.xlsx')
df_ATP_ODDS = df_ATP_ODDS.filter(['tourney_id','tourney_date','match_num',
'round','winner_id','loser_id','surface','tourney_level','best_of','AvgW','AvgL'])

# Adiciona a coluna winner para indicar que foi o vencedor : jogador 1 ou 2
# -----

# df_ATP_ODDS['winner'] = df_ATP_ODDS.apply(lambda row: select_winner
(row), axis=1)

# Adiciona as colunas winner, player_1 e player_2 para poder fazer o merge
com os dados agregados do par de jogadores
# -----
-----

# df_ATP_ODDS[['player_1', 'player_2']] = df_ATP_ODDS.apply(lambda row:
select_player_1_2 (row), axis=1)
# df_ATP_ODDS['player_1'] = df_ATP_ODDS.apply(lambda row: select_player_1
(row), axis=1)
# df_ATP_ODDS['player_2'] = df_ATP_ODDS.apply(lambda row: select_player_2
(row), axis=1)

df_ATP_ODDS=df_ATP_ODDS.apply(select_winner_player_1_2,axis=1)

# Elimina linhas com ODDS nulos
# -----

df_ATP_ODDS = df_ATP_ODDS[df_ATP_ODDS['AvgW'].notnull()]

```



```

df_ATP_ODDS = df_ATP_ODDS[df_ATP_ODDS['AvgL'].notnull()]

print(df_ATP_ODDS)

df_ATP_ODDS.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python
\Saved_2\merged_ATP_ODDS_final.csv', index = False)

def gera_bases_finais():

    # Obtem os dados de merge de resultados da ATP e odds de apostas
    # -----

    df_ATP_ODDS =
pd.read_csv('C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\Saved_2\
merged_ATP_ODDS_final.csv', sep=",")

    # Obtem os dados agregados de cada jogador
    # -----

    df1_player =
pd.read_csv('C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\Saved_2\
df1_player_geral.csv', sep=";")
    df1_player_aux = df1_player

    #      Altera o número de sequência dos jogos para que os dados agregados
    #      fiquem associados às partidas seguintes
    #      -----
    ---

    df1_player_aux.match_seq = df1_player.match_seq + 1

    #      Obtem as colunas relevantes

```

```

# -----

df1_player_aux.merge(df1_player, how='left', left_on=['player_id', 'match_seq'],
right_on=['player_id', 'match_seq'])
df1_player_aux = df1_player_aux.filter(['player_id', 'tourney_id', 'tourney_date',
'match_num',
'round','pct_grass_victories','pct_hard_victories','pct_clay_victories','pct_3sets_victori
es','pct_5sets_victories','avg_ace','avg_df','avg_svpt','avg_1stIn','avg_1stWon','avg_2
ndWon','avg_SvGms','avg_bpSaved','avg_bpFaced','age','rank_points','pct_victories_
grand_slam','pct_victories_master_1000','pct_victories_ATP_250_500','pct_victories_
finals','qty_grand_slam_titles','qty_master_1000_titles','qty_ATP_250_500_titles','qty
_finals_titles'])

df1_player_aux.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python
\df1_player_aux_XX.csv', index = False, sep=';')

# Obtem os dados agregados de cada par de jogadores
# -----

df2_player =
pd.read_csv('C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\Saved_2\
df2_player_geral.csv', sep=";")
df2_player_aux = df2_player

#      Altera o número de sequência dos jogos para que os dados agregados
fiquem associados às partidas seguintes
# -----
---

df2_player_aux.match_seq = df2_player.match_seq + 1

#      Obtem as colunas relevantes
# -----

```

```

df2_player_aux.merge(df2_player, how='left', left_on=['player_1_id', 'player_2_id',
'match_seq'], right_on=['player_1_id', 'player_2_id', 'match_seq'])
df2_player_aux = df2_player_aux.filter(['player_1_id', 'player_2_id', 'tourney_id',
'tourney_date', 'match_num',
'round','pct_player_1_grass_victories','pct_player_1_hard_victories','pct_player_1_cla
y_victories','pct_player_1_3sets_victories','pct_player_1_5sets_victories','pct_player_
1_grand_slam_victories','pct_player_1_master_1000_victories','pct_player_1_finals_
victories','pct_player_1_ATP_250_500_victories','qty_grass_matches','qty_hard_mat
ches','qty_clay_matches','qty_3sets_matches','qty_5sets_matches','qty_grand_slam_
matches','qty_master_1000_matches','qty_finals_matches','qty_ATP_250_500_matc
hes'

])

```

```

df2_player_aux.to_csv(r'C:\Giba\Dados\A_Giba\Documentos\Adm\PUC_MG\Python
\df2_player_aux_XX.csv', index = False, sep=';')

```

```

# Merge com dados agregados do winner
# -----

```

```

df_merge_all = df_ATP_ODDS.merge(df1_player_aux, how='left',
left_on=['winner_id', 'tourney_id', 'tourney_date', 'match_num', 'round'],
right_on=['player_id', 'tourney_id', 'tourney_date', 'match_num', 'round'])

```

```

# Merge com dados agregados do loser
# -----

```

```

df_merge_all = df_merge_all.merge(df1_player_aux, how='left', left_on=['loser_id',
'tourney_id', 'tourney_date', 'match_num', 'round'], right_on=['player_id', 'tourney_id',
'tourney_date', 'match_num', 'round'])

```

```

# Merge com dados agregados do par de jogadores
# -----

```

```
df_merge_all = df_merge_all.merge(df2_player_aux, how='left',
left_on=['player_1_id', 'player_2_id', 'tourney_id', 'tourney_date', 'match_num',
'round'], right_on=['player_1_id', 'player_2_id', 'tourney_id', 'tourney_date',
'match_num', 'round'])
```

```
df_merge_all.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\
merge_all_XX.csv', index = False , sep=';')
```

```
# Gera base final completa
```

```
# -----
```

```
columns = {
    'tourney_id':",
    'tourney_date':",
    'match_num':int,
    'round':",
    'surface':",
    'tourney_level':",
    'best_of':int,
    'AvgW':float,
    'AvgL':float,

    'player_id_1':int,
    'pct_surface_victories_1':float,
    'pct_best_of_victories_1':float,
    'avg_ace_1':float,
    'avg_df_1':float,
    'avg_svpt_1':float,
    'avg_1stIn_1':float,
    'avg_1stWon_1':float,
    'avg_2ndWon_1':float,
    'avg_SvGms_1':float,
```

'avg\_bpSaved\_1':float,  
'avg\_bpFaced\_1':float,  
'age\_1':int,  
'rank\_points\_1':int,  
'pct\_victories\_grand\_slam\_1':float,  
'pct\_victories\_master\_1000\_1':float,  
'pct\_victories\_ATP\_250\_500\_1':float,  
'pct\_victories\_finals\_1':float,  
'qty\_grand\_slam\_titles\_1':int,  
'qty\_master\_1000\_titles\_1':int,  
'qty\_ATP\_250\_500\_titles\_1':int,  
'qty\_finals\_titles\_1':int,  
  
'player\_id\_2':int,  
'pct\_surface\_victories\_2':float,  
'pct\_best\_of\_victories\_2':float,  
'avg\_ace\_2':float,  
'avg\_df\_2':float,  
'avg\_svpt\_2':float,  
'avg\_1stIn\_2':float,  
'avg\_1stWon\_2':float,  
'avg\_2ndWon\_2':float,  
'avg\_SvGms\_2':float,  
'avg\_bpSaved\_2':float,  
'avg\_bpFaced\_2':float,  
'age\_2':int,  
'rank\_points\_2':int,  
'pct\_victories\_grand\_slam\_2':float,  
'pct\_victories\_master\_1000\_2':float,  
'pct\_victories\_ATP\_250\_500\_2':float,  
'pct\_victories\_finals\_2':float,  
'qty\_grand\_slam\_titles\_2':int,  
'qty\_master\_1000\_titles\_2':int,  
'qty\_ATP\_250\_500\_titles\_2':int,

```

'qty_finals_titles_2':int,

'pct_player_1_surface_victories':float,
'pct_player_1_best_of_victories':float,
'pct_player_1_grand_slam_victories':float,
'pct_player_1_master_1000_victories':float,
'pct_player_1_finals_victories':float,
'pct_player_1_ATP_250_500_victories':float,

'pct_player_2_surface_victories':float,
'pct_player_2_best_of_victories':float,
'pct_player_2_grand_slam_victories':float,
'pct_player_2_master_1000_victories':float,
'pct_player_2_finals_victories':float,
'pct_player_2_ATP_250_500_victories':float,

'winner':int
}

```

```

contador = 0

```

```

df_base_completa = pd.DataFrame(columns, index=[])

```

```

for ind in df_merge_all.index:

```

```

    contador = contador + 1

```

```

    if (contador % 100) == 1:

```

```

        now = datetime.now()

```

```

        print(str(contador) + '-' + now.strftime("%H:%M:%S"))

```

```

    # if contador == 302:

```

```

    #     break

```

```
# print('250: ' + df_merge_all.loc[ind,'pct_player_1_ATP_250_500_victories'])
```

```
df_base_completa_aux = pd.DataFrame([
    df_merge_all.loc[ind,'tourney_id'],
    df_merge_all.loc[ind,'tourney_date'],
    df_merge_all.loc[ind,'match_num'],
    df_merge_all.loc[ind,'round'],
    df_merge_all.loc[ind,'surface'],
    df_merge_all.loc[ind,'tourney_level'],
    df_merge_all.loc[ind,'best_of'],
    df_merge_all.loc[ind,'AvgW'],
    df_merge_all.loc[ind,'AvgL'],

    0, # player_id_1
    0, # pct_surface_victories_1
    0, # pct_best_of_victories_1
    0, # avg_ace_1
    0, # avg_df_1
    0, # avg_svpt_1
    0, # avg_1stIn_1
    0, # avg_1stWon_1
    0, # avg_2ndWon_1
    0, # avg_SvGms_1
    0, # avg_bpSaved_1
    0, # avg_bpFaced_1
    0, # age_1
    0, # rank_points_1
    0, # pct_victories_grand_slam_1
    0, # pct_victories_master_1000_1
    0, # pct_victories_ATP_250_500_1
    0, # pct_victories_finals_1
    0, # qty_grand_slam_titles_1
    0, # qty_master_1000_titles_1
    0, # qty_ATP_250_500_titles_1
```

0, # qty\_finals\_titles\_1

0, # player\_id\_2

0, # pct\_surface\_victories\_2

0, # pct\_best\_of\_victories\_2

0, # avg\_ace\_2

0, # avg\_df\_2

0, # avg\_svpt\_2

0, # avg\_1stIn\_2

0, # avg\_1stWon\_2

0, # avg\_2ndWon\_2

0, # avg\_SvGms\_2

0, # avg\_bpSaved\_2

0, # avg\_bpFaced\_2

0, # age\_2

0, # rank\_points\_2

0, # pct\_victories\_grand\_slam\_2

0, # pct\_victories\_master\_1000\_2

0, # pct\_victories\_ATP\_250\_500\_2

0, # pct\_victories\_finals\_2

0, # qty\_grand\_slam\_titles\_2

0, # qty\_master\_1000\_titles\_2

0, # qty\_ATP\_250\_500\_titles\_2

0, # qty\_finals\_titles\_2

0, # pct\_player\_1\_surface\_victories

0, # pct\_player\_1\_best\_of\_victories

0, # pct\_player\_1\_grand\_slam\_victories

0, # pct\_player\_1\_master\_1000\_victories

0, # pct\_player\_1\_finals\_victories

0, # pct\_player\_1\_ATP\_250\_500\_victories

0, # pct\_player\_2\_surface\_victories

0, # pct\_player\_2\_best\_of\_victories



```

0, # pct_player_2_grand_slam_victories
0, # pct_player_2_master_1000_victories
0, # pct_player_2_finals_victories
0, # pct_player_2_ATP_250_500_victories

df_merge_all.loc[ind,'winner']

    ], columns = columns)

if df_merge_all.loc[ind,'winner'] == 1:

    # Player_1
    # -----

    df_base_completa_aux.loc[0,"player_id_1"] =
df_merge_all.loc[ind,"player_id_x"]

    if df_merge_all.loc[ind,'surface'] == 'hard':
        df_base_completa_aux.loc[0,"pct_surface_victories_1"] =
df_merge_all.loc[ind,"pct_hard_victories_x"]
    elif df_merge_all.loc[ind,'surface'] == 'clay':
        df_base_completa_aux.loc[0,"pct_surface_victories_1"] =
df_merge_all.loc[ind,"pct_clay_victories_x"]
    else:
        df_base_completa_aux.loc[0,"pct_surface_victories_1"] =
df_merge_all.loc[ind,"pct_grass_victories_x"]

    if df_merge_all.loc[ind,'best_of'] == 3:
        df_base_completa_aux.loc[0,"pct_best_of_victories_1"] =
df_merge_all.loc[ind,"pct_3sets_victories_x"]
    else:
        df_base_completa_aux.loc[0,"pct_best_of_victories_1"] =
df_merge_all.loc[ind,"pct_5sets_victories_x"]

```

```

df_base_completa_aux.loc[0,"avg_ace_1"] =
df_merge_all.loc[ind,"avg_ace_x"]
df_base_completa_aux.loc[0,"avg_df_1"] = df_merge_all.loc[ind,"avg_df_x"]
df_base_completa_aux.loc[0,"avg_svpt_1"] =
df_merge_all.loc[ind,"avg_svpt_x"]
df_base_completa_aux.loc[0,"avg_1stln_1"] =
df_merge_all.loc[ind,"avg_1stln_x"]
df_base_completa_aux.loc[0,"avg_1stWon_1"] =
df_merge_all.loc[ind,"avg_1stWon_x"]
df_base_completa_aux.loc[0,"avg_2ndWon_1"] =
df_merge_all.loc[ind,"avg_2ndWon_x"]
df_base_completa_aux.loc[0,"avg_SvGms_1"] =
df_merge_all.loc[ind,"avg_SvGms_x"]
df_base_completa_aux.loc[0,"avg_bpSaved_1"] =
df_merge_all.loc[ind,"avg_bpSaved_x"]
df_base_completa_aux.loc[0,"avg_bpFaced_1"] =
df_merge_all.loc[ind,"avg_bpFaced_x"]
df_base_completa_aux.loc[0,"age_1"] = df_merge_all.loc[ind,"age_x"]
df_base_completa_aux.loc[0,"rank_points_1"] =
df_merge_all.loc[ind,"rank_points_x"]
df_base_completa_aux.loc[0,"pct_victories_grand_slam_1"] =
df_merge_all.loc[ind,"pct_victories_grand_slam_x"]
df_base_completa_aux.loc[0,"pct_victories_master_1000_1"] =
df_merge_all.loc[ind,"pct_victories_master_1000_x"]
df_base_completa_aux.loc[0,"pct_victories_ATP_250_500_1"] =
df_merge_all.loc[ind,"pct_victories_ATP_250_500_x"]
df_base_completa_aux.loc[0,"pct_victories_finals_1"] =
df_merge_all.loc[ind,"pct_victories_finals_x"]
df_base_completa_aux.loc[0,"qty_grand_slam_titles_1"] =
df_merge_all.loc[ind,"qty_grand_slam_titles_x"]
df_base_completa_aux.loc[0,"qty_master_1000_titles_1"] =
df_merge_all.loc[ind,"qty_master_1000_titles_x"]
df_base_completa_aux.loc[0,"qty_ATP_250_500_titles_1"] =
df_merge_all.loc[ind,"qty_ATP_250_500_titles_x"]

```

```

df_base_completa_aux.loc[0,"qty_finals_titles_1"] =
df_merge_all.loc[ind,"qty_finals_titles_x"]

# Player_2
# -----

df_base_completa_aux.loc[0,"player_id_2"] =
df_merge_all.loc[ind,"player_id_y"]

if df_merge_all.loc[ind,'surface'] == 'hard':
    df_base_completa_aux.loc[0,"pct_surface_victories_2"] =
df_merge_all.loc[ind,"pct_hard_victories_y"]
    elif df_merge_all.loc[ind,'surface'] == 'clay':
        df_base_completa_aux.loc[0,"pct_surface_victories_2"] =
df_merge_all.loc[ind,"pct_clay_victories_y"]
    else:
        df_base_completa_aux.loc[0,"pct_surface_victories_2"] =
df_merge_all.loc[ind,"pct_grass_victories_y"]

if df_merge_all.loc[ind,'best_of'] == 3:
    df_base_completa_aux.loc[0,"pct_best_of_victories_2"] =
df_merge_all.loc[ind,"pct_3sets_victories_y"]
    else:
        df_base_completa_aux.loc[0,"pct_best_of_victories_2"] =
df_merge_all.loc[ind,"pct_5sets_victories_y"]

df_base_completa_aux.loc[0,"avg_ace_2"] =
df_merge_all.loc[ind,"avg_ace_y"]
    df_base_completa_aux.loc[0,"avg_df_2"] = df_merge_all.loc[ind,"avg_df_y"]
    df_base_completa_aux.loc[0,"avg_svpt_2"] =
df_merge_all.loc[ind,"avg_svpt_y"]
    df_base_completa_aux.loc[0,"avg_1stln_2"] =
df_merge_all.loc[ind,"avg_1stln_y"]

```

```

        df_base_completa_aux.loc[0,"avg_1stWon_2"] =
df_merge_all.loc[ind,"avg_1stWon_y"]
        df_base_completa_aux.loc[0,"avg_2ndWon_2"] =
df_merge_all.loc[ind,"avg_2ndWon_y"]
        df_base_completa_aux.loc[0,"avg_SvGms_2"] =
df_merge_all.loc[ind,"avg_SvGms_y"]
        df_base_completa_aux.loc[0,"avg_bpSaved_2"] =
df_merge_all.loc[ind,"avg_bpSaved_y"]
        df_base_completa_aux.loc[0,"avg_bpFaced_2"] =
df_merge_all.loc[ind,"avg_bpFaced_y"]
        df_base_completa_aux.loc[0,"age_2"] = df_merge_all.loc[ind,"age_y"]
        df_base_completa_aux.loc[0,"rank_points_2"] =
df_merge_all.loc[ind,"rank_points_y"]
        df_base_completa_aux.loc[0,"pct_victories_grand_slam_2"] =
df_merge_all.loc[ind,"pct_victories_grand_slam_y"]
        df_base_completa_aux.loc[0,"pct_victories_master_1000_2"] =
df_merge_all.loc[ind,"pct_victories_master_1000_y"]
        df_base_completa_aux.loc[0,"pct_victories_ATP_250_500_2"] =
df_merge_all.loc[ind,"pct_victories_ATP_250_500_y"]
        df_base_completa_aux.loc[0,"pct_victories_finals_2"] =
df_merge_all.loc[ind,"pct_victories_finals_y"]
        df_base_completa_aux.loc[0,"qty_grand_slam_titles_2"] =
df_merge_all.loc[ind,"qty_grand_slam_titles_y"]
        df_base_completa_aux.loc[0,"qty_master_1000_titles_2"] =
df_merge_all.loc[ind,"qty_master_1000_titles_y"]
        df_base_completa_aux.loc[0,"qty_ATP_250_500_titles_2"] =
df_merge_all.loc[ind,"qty_ATP_250_500_titles_y"]
        df_base_completa_aux.loc[0,"qty_finals_titles_2"] =
df_merge_all.loc[ind,"qty_finals_titles_y"]

```

else:

# Player\_1

# -----

```

df_base_completa_aux.loc[0,"player_id_1"] =
df_merge_all.loc[ind,"player_id_y"]

if df_merge_all.loc[ind,'surface'] == 'hard':
    df_base_completa_aux.loc[0,"pct_surface_victories_1"] =
df_merge_all.loc[ind,"pct_hard_victories_y"]
    elif df_merge_all.loc[ind,'surface'] == 'clay':
        df_base_completa_aux.loc[0,"pct_surface_victories_1"] =
df_merge_all.loc[ind,"pct_clay_victories_y"]
    else:
        df_base_completa_aux.loc[0,"pct_surface_victories_1"] =
df_merge_all.loc[ind,"pct_grass_victories_y"]

if df_merge_all.loc[ind,'best_of'] == 3:
    df_base_completa_aux.loc[0,"pct_best_of_victories_1"] =
df_merge_all.loc[ind,"pct_3sets_victories_y"]
    else:
        df_base_completa_aux.loc[0,"pct_best_of_victories_1"] =
df_merge_all.loc[ind,"pct_5sets_victories_y"]

df_base_completa_aux.loc[0,"avg_ace_1"] =
df_merge_all.loc[ind,"avg_ace_y"]
    df_base_completa_aux.loc[0,"avg_df_1"] = df_merge_all.loc[ind,"avg_df_y"]
    df_base_completa_aux.loc[0,"avg_svpt_1"] =
df_merge_all.loc[ind,"avg_svpt_y"]
    df_base_completa_aux.loc[0,"avg_1stln_1"] =
df_merge_all.loc[ind,"avg_1stln_y"]
    df_base_completa_aux.loc[0,"avg_1stWon_1"] =
df_merge_all.loc[ind,"avg_1stWon_y"]
    df_base_completa_aux.loc[0,"avg_2ndWon_1"] =
df_merge_all.loc[ind,"avg_2ndWon_y"]
    df_base_completa_aux.loc[0,"avg_SvGms_1"] =
df_merge_all.loc[ind,"avg_SvGms_y"]

```

```

        df_base_completa_aux.loc[0,"avg_bpSaved_1"] =
df_merge_all.loc[ind,"avg_bpSaved_y"]
        df_base_completa_aux.loc[0,"avg_bpFaced_1"] =
df_merge_all.loc[ind,"avg_bpFaced_y"]
        df_base_completa_aux.loc[0,"age_1"] = df_merge_all.loc[ind,"age_y"]
        df_base_completa_aux.loc[0,"rank_points_1"] =
df_merge_all.loc[ind,"rank_points_y"]
        df_base_completa_aux.loc[0,"pct_victories_grand_slam_1"] =
df_merge_all.loc[ind,"pct_victories_grand_slam_y"]
        df_base_completa_aux.loc[0,"pct_victories_master_1000_1"] =
df_merge_all.loc[ind,"pct_victories_master_1000_y"]
        df_base_completa_aux.loc[0,"pct_victories_ATP_250_500_1"] =
df_merge_all.loc[ind,"pct_victories_ATP_250_500_y"]
        df_base_completa_aux.loc[0,"pct_victories_finals_1"] =
df_merge_all.loc[ind,"pct_victories_finals_y"]
        df_base_completa_aux.loc[0,"qty_grand_slam_titles_1"] =
df_merge_all.loc[ind,"qty_grand_slam_titles_y"]
        df_base_completa_aux.loc[0,"qty_master_1000_titles_1"] =
df_merge_all.loc[ind,"qty_master_1000_titles_y"]
        df_base_completa_aux.loc[0,"qty_ATP_250_500_titles_1"] =
df_merge_all.loc[ind,"qty_ATP_250_500_titles_y"]
        df_base_completa_aux.loc[0,"qty_finals_titles_1"] =
df_merge_all.loc[ind,"qty_finals_titles_y"]

# Player_2
# -----

        df_base_completa_aux.loc[0,"player_id_2"] =
df_merge_all.loc[ind,"player_id_x"]

        if df_merge_all.loc[ind,'surface'] == 'hard':
            df_base_completa_aux.loc[0,"pct_surface_victories_2"] =
df_merge_all.loc[ind,"pct_hard_victories_x"]
            elif df_merge_all.loc[ind,'surface'] == 'clay':

```

```

        df_base_completa_aux.loc[0,"pct_surface_victories_2"] =
df_merge_all.loc[ind,"pct_clay_victories_x"]
    else:
        df_base_completa_aux.loc[0,"pct_surface_victories_2"] =
df_merge_all.loc[ind,"pct_grass_victories_x"]

    if df_merge_all.loc[ind,'best_of'] == 3:
        df_base_completa_aux.loc[0,"pct_best_of_victories_2"] =
df_merge_all.loc[ind,"pct_3sets_victories_x"]
    else:
        df_base_completa_aux.loc[0,"pct_best_of_victories_2"] =
df_merge_all.loc[ind,"pct_5sets_victories_x"]

    df_base_completa_aux.loc[0,"avg_ace_2"] =
df_merge_all.loc[ind,"avg_ace_x"]
    df_base_completa_aux.loc[0,"avg_df_2"] = df_merge_all.loc[ind,"avg_df_x"]
    df_base_completa_aux.loc[0,"avg_svpt_2"] =
df_merge_all.loc[ind,"avg_svpt_x"]
    df_base_completa_aux.loc[0,"avg_1stln_2"] =
df_merge_all.loc[ind,"avg_1stln_x"]
    df_base_completa_aux.loc[0,"avg_1stWon_2"] =
df_merge_all.loc[ind,"avg_1stWon_x"]
    df_base_completa_aux.loc[0,"avg_2ndWon_2"] =
df_merge_all.loc[ind,"avg_2ndWon_x"]
    df_base_completa_aux.loc[0,"avg_SvGms_2"] =
df_merge_all.loc[ind,"avg_SvGms_x"]
    df_base_completa_aux.loc[0,"avg_bpSaved_2"] =
df_merge_all.loc[ind,"avg_bpSaved_x"]
    df_base_completa_aux.loc[0,"avg_bpFaced_2"] =
df_merge_all.loc[ind,"avg_bpFaced_x"]
    df_base_completa_aux.loc[0,"age_2"] = df_merge_all.loc[ind,"age_x"]
    df_base_completa_aux.loc[0,"rank_points_2"] =
df_merge_all.loc[ind,"rank_points_x"]

```

```

        df_base_completa_aux.loc[0,"pct_victories_grand_slam_2"] =
df_merge_all.loc[ind,"pct_victories_grand_slam_x"]
        df_base_completa_aux.loc[0,"pct_victories_master_1000_2"] =
df_merge_all.loc[ind,"pct_victories_master_1000_x"]
        df_base_completa_aux.loc[0,"pct_victories_ATP_250_500_2"] =
df_merge_all.loc[ind,"pct_victories_ATP_250_500_x"]
        df_base_completa_aux.loc[0,"pct_victories_finals_2"] =
df_merge_all.loc[ind,"pct_victories_finals_x"]
        df_base_completa_aux.loc[0,"qty_grand_slam_titles_2"] =
df_merge_all.loc[ind,"qty_grand_slam_titles_x"]
        df_base_completa_aux.loc[0,"qty_master_1000_titles_2"] =
df_merge_all.loc[ind,"qty_master_1000_titles_x"]
        df_base_completa_aux.loc[0,"qty_ATP_250_500_titles_2"] =
df_merge_all.loc[ind,"qty_ATP_250_500_titles_x"]
        df_base_completa_aux.loc[0,"qty_finals_titles_2"] =
df_merge_all.loc[ind,"qty_finals_titles_x"]

    if df_merge_all.loc[ind,'surface'] == 'hard':
        if df_merge_all.loc[ind,'qty_hard_matches'] > 0:
            df_base_completa_aux.loc[0,"pct_player_1_surface_victories"] =
df_merge_all.loc[ind,"pct_player_1_grass_victories"]
            df_base_completa_aux.loc[0,"pct_player_2_surface_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_grass_victories"].replace(",","."))
        else:
            df_base_completa_aux.loc[0,"pct_player_1_surface_victories"] = 0
            df_base_completa_aux.loc[0,"pct_player_2_surface_victories"] = 0

    elif df_merge_all.loc[ind,'surface'] == 'clay':
        if df_merge_all.loc[ind,'qty_clay_matches'] > 0:
            df_base_completa_aux.loc[0,"pct_player_1_surface_victories"] =
df_merge_all.loc[ind,"pct_player_1_clay_victories"]
            df_base_completa_aux.loc[0,"pct_player_2_surface_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_clay_victories"].replace(",","."))
        else:

```



```

df_base_completa_aux.loc[0,"pct_player_1_surface_victories"] = 0
df_base_completa_aux.loc[0,"pct_player_2_surface_victories"] = 0
else:
    if df_merge_all.loc[ind,'qty_grass_matches'] > 0:
        df_base_completa_aux.loc[0,"pct_player_1_surface_victories"] =
df_merge_all.loc[ind,"pct_player_1_grass_victories"]
        df_base_completa_aux.loc[0,"pct_player_2_surface_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_grass_victories"].replace(",","."))
    else:
        df_base_completa_aux.loc[0,"pct_player_1_surface_victories"] = 0
        df_base_completa_aux.loc[0,"pct_player_2_surface_victories"] = 0

    if df_merge_all.loc[ind,'best_of'] == 3:
        if df_merge_all.loc[ind,'qty_3sets_matches'] > 0:
            df_base_completa_aux.loc[0,"pct_player_1_best_of_victories"] =
df_merge_all.loc[ind,"pct_player_1_3sets_victories"]
            df_base_completa_aux.loc[0,"pct_player_2_best_of_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_3sets_victories"].replace(",","."))
        else:
            df_base_completa_aux.loc[0,"pct_player_1_best_of_victories"] = 0
            df_base_completa_aux.loc[0,"pct_player_2_best_of_victories"] = 0
    else:
        if df_merge_all.loc[ind,'qty_5sets_matches'] > 0:
            df_base_completa_aux.loc[0,"pct_player_1_best_of_victories"] =
df_merge_all.loc[ind,"pct_player_1_5sets_victories"]
            df_base_completa_aux.loc[0,"pct_player_2_best_of_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_5sets_victories"].replace(",","."))
        else:
            df_base_completa_aux.loc[0,"pct_player_1_best_of_victories"] = 0
            df_base_completa_aux.loc[0,"pct_player_2_best_of_victories"] = 0

    if df_merge_all.loc[ind,'qty_grand_slam_matches'] > 0:

```

```

df_base_completa_aux.loc[0,"pct_player_1_grand_slam_victories"] =
df_merge_all.loc[ind,"pct_player_1_grand_slam_victories"]
df_base_completa_aux.loc[0,"pct_player_2_grand_slam_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_grand_slam_victories"].replace(",","."))
else:
df_base_completa_aux.loc[0,"pct_player_1_grand_slam_victories"] = 0
df_base_completa_aux.loc[0,"pct_player_2_grand_slam_victories"] = 0

if df_merge_all.loc[ind,'qty_master_1000_matches'] > 0:
df_base_completa_aux.loc[0,"pct_player_1_master_1000_victories"] =
df_merge_all.loc[ind,"pct_player_1_master_1000_victories"]
df_base_completa_aux.loc[0,"pct_player_2_master_1000_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_master_1000_victories"].replace(",","."))
else:
df_base_completa_aux.loc[0,"pct_player_1_master_1000_victories"] = 0
df_base_completa_aux.loc[0,"pct_player_2_master_1000_victories"] = 0

if df_merge_all.loc[ind,'qty_finals_matches'] > 0:
df_base_completa_aux.loc[0,"pct_player_1_finals_victories"] =
df_merge_all.loc[ind,"pct_player_1_finals_victories"]
df_base_completa_aux.loc[0,"pct_player_2_finals_victories"] = 100 -
float(df_merge_all.loc[ind,"pct_player_1_finals_victories"].replace(",","."))
else:
df_base_completa_aux.loc[0,"pct_player_1_finals_victories"] = 0
df_base_completa_aux.loc[0,"pct_player_2_finals_victories"] = 0

if df_merge_all.loc[ind,'qty_ATP_250_500_matches'] > 0:
df_base_completa_aux.loc[0,"pct_player_1_ATP_250_500_victories"] =
df_merge_all.loc[ind,"pct_player_1_ATP_250_500_victories"]
df_base_completa_aux.loc[0,"pct_player_2_ATP_250_500_victories"] = 100
- float(df_merge_all.loc[ind,"pct_player_1_ATP_250_500_victories"].replace(",","."))
else:
df_base_completa_aux.loc[0,"pct_player_1_ATP_250_500_victories"] = 0
df_base_completa_aux.loc[0,"pct_player_2_ATP_250_500_victories"] = 0

```

```
df_base_completa = df_base_completa.append(df_base_completa_aux.iloc[0])  
del df_base_completa_aux
```

```
df_base_completa.to_csv(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\base_completa_XX.csv', index = False , sep=';')
```

```
df_base_completa.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\base_completa_XX.xlsx")
```

```
print('OK')
```

## TCC\_PUC-MG\_Ciencia-de-Dados-e-Big-Data\_2019\_Gilberto\_Klingen - Parte 2 - Modelos de Machine Learning.ipynb

```
#!/usr/bin/env python
```

```
# coding: utf-8
```

```
# ## PUC - MG
```

```
# ## Ciência de Dados e Big Data - 2019
```

```
# ## TCC - Gilberto Klingen
```

```
import pandas as pd
```

```
import numpy as np
```

```
from sklearn import datasets, tree
```

```
from sklearn.model_selection import train_test_split
```

```
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
```

```
from sklearn.feature_extraction import DictVectorizer
```

```
from sklearn.preprocessing import LabelEncoder
```

```
# ## Aquisição de Dados
```

```
def obtem_base_reduzida(base):
```

```
    # Esta função transforma duas colunas de mesmo significado em apenas uma  
    contendo a diferença do valor relativo
```

```
    # ao jogador 1 diminuída do valor relativo ao jogador 2
```

```
    base['pct_surface_victories_1'] = base['pct_surface_victories_1'] -  
    base['pct_surface_victories_2']
```

```
    base['pct_best_of_victories_1'] = base['pct_best_of_victories_1'] -  
    base['pct_best_of_victories_2']
```

```
    base['avg_ace_1'] = base['avg_ace_1'] - base['avg_ace_2']
```

```

base['avg_df_1'] = base['avg_df_1'] - base['avg_df_2']
base['avg_svpt_1'] = base['avg_svpt_1'] - base['avg_svpt_2']
base['avg_1stIn_1'] = base['avg_1stIn_1'] - base['avg_1stIn_2']
base['avg_1stWon_1'] = base['avg_1stWon_1'] - base['avg_1stWon_2']
base['avg_2ndWon_1'] = base['avg_2ndWon_1'] - base['avg_2ndWon_2']
base['avg_SvGms_1'] = base['avg_SvGms_1'] - base['avg_SvGms_2']
base['avg_bpSaved_1'] = base['avg_bpSaved_1'] - base['avg_bpSaved_2']
base['avg_bpFaced_1'] = base['avg_bpFaced_1'] - base['avg_bpFaced_2']
base['age_1'] = base['age_1'] - base['age_2']
base['rank_points_1'] = base['rank_points_1'] - base['rank_points_2']
base['pct_victories_grand_slam_1']    =    base['pct_victories_grand_slam_1']    -
base['pct_victories_grand_slam_2']
base['pct_victories_master_1000_1']  =    base['pct_victories_master_1000_1']  -
base['pct_victories_master_1000_2']
base['pct_victories_ATP_250_500_1']  =    base['pct_victories_ATP_250_500_1']  -
base['pct_victories_ATP_250_500_2']
base['pct_victories_finals_1']        =        base['pct_victories_finals_1']        -
base['pct_victories_finals_2']
base['qty_grand_slam_titles_1']       =       base['qty_grand_slam_titles_1']       -
base['qty_grand_slam_titles_2']
base['qty_master_1000_titles_1']      =      base['qty_master_1000_titles_1']      -
base['qty_master_1000_titles_2']
base['qty_ATP_250_500_titles_1']      =      base['qty_ATP_250_500_titles_1']      -
base['qty_ATP_250_500_titles_2']
base['qty_finals_titles_1'] = base['qty_finals_titles_1'] - base['qty_finals_titles_2']

base['pct_player_1_surface_victories'] = base['pct_player_1_surface_victories'] -
base['pct_player_2_surface_victories']
base['pct_player_1_best_of_victories'] = base['pct_player_1_best_of_victories'] -
base['pct_player_2_best_of_victories']
base['pct_player_1_grand_slam_victories'] =
base['pct_player_1_grand_slam_victories'] -
base['pct_player_2_grand_slam_victories']

```

```

    base['pct_player_1_master_1000_victories'] =
base['pct_player_1_master_1000_victories'] -
base['pct_player_2_master_1000_victories']
    base['pct_player_1_finals_victories'] = base['pct_player_1_finals_victories'] -
base['pct_player_2_finals_victories']
    base['pct_player_1_ATP_250_500_victories'] =
base['pct_player_1_ATP_250_500_victories'] -
base['pct_player_2_ATP_250_500_victories']

    base =
base.drop(['pct_surface_victories_2','pct_best_of_victories_2','avg_ace_2','avg_df_2',
'avg_svpt_2','avg_1stIn_2',

'avg_1stWon_2','avg_2ndWon_2','avg_SvGms_2','avg_bpSaved_2','avg_bpFaced_2'
,'age_2','rank_points_2',

'pct_victories_grand_slam_2','pct_victories_master_1000_2','pct_victories_ATP_250
_500_2',

'pct_victories_finals_2','qty_grand_slam_titles_2','qty_master_1000_titles_2','qty_AT
P_250_500_titles_2',

'qty_finals_titles_2','pct_player_2_surface_victories','pct_player_2_best_of_victories',

'pct_player_2_grand_slam_victories','pct_player_2_master_1000_victories','pct_play
er_2_finals_victories',
        'pct_player_2_ATP_250_500_victories'],axis=1)

    return base

```

# Obtem a base completa de jogos para análise

```
base_completa =
pd.read_excel(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Python\Bases_
ML\base_completa.xlsx')
```

# Gera a base completa redefinida, onde os valores irão de 0,0 a 1,0 e exclui as colunas não relevantes

```
base_redefinida =
base_completa.drop(['tourney_id','tourney_date','match_num','round','surface','tourne
y_level',
                    'best_of','AvgW','AvgL','player_id_1','player_id_2'],axis=1)
base_redefinida = base_redefinida.apply(lambda x: round(x / 100,4))
base_redefinida['rank_points_1'] = base_redefinida['rank_points_1'].apply(lambda x:
round(x / 1000,4))
base_redefinida['rank_points_2'] = base_redefinida['rank_points_2'].apply(lambda x:
round(x / 1000,4))
base_redefinida['winner'] = base_redefinida['winner'].apply(lambda x: x * 100)
base_redefinida.to_excel(r'C:\Giba\Dados\_A_Giba\Documentos\Adm\PUC_MG\Pyth
on\Bases_ML\base_redefinida.xlsx')
```

# Gera a base completa reduzida, onde as colunas do jogador 1 e 2 foram reduzidas a apenas uma contendo a diferença

# de seus valores

```
base_ML = obtem_base_reduzida(base_redefinida)
```

# ## Gera o Modelo de Machine Learning utilizando o algoritmo de regressão logística

```
from sklearn.linear_model import LogisticRegression
```

```
# Separa as colunas de dimensões da base
```

```
X = base_ML.iloc[:,0:(base_ML.shape[1] - 1)]
```

```
# Separa a coluna de classe de classificação da base
```

```
le = LabelEncoder()
```

```
y = le.fit_transform(base_ML.iloc[:,(base_ML.shape[1] - 1)])
```

```
# Faz a separação das bases de treinamento e teste (20% da base será para  
treinamento)
```

```
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=0  
)
```

```
# Executa o treinamento do modelo
```

```
LogisticRegr = LogisticRegression(max_iter=1000)
```

```
LogisticRegr.fit(x_train, y_train)
```

```
# Aplica o modelo na base de teste
```

```
y_pred = LogisticRegr.predict(x_test)
```

```
# Verifica a acurácia do modelo
```

```
score = accuracy_score(y_test, y_pred)
```



```
print(score)
```

```
# Verifica a performance de classificação do modelo
```

```
print(classification_report( y_test, y_pred))
```

```
# Verifica a matriz de confusão do modelo
```

```
confusion = confusion_matrix(y_test, y_pred)
```

```
cnf_table = pd.DataFrame(data=confusion, index=["Player_1", "Player_2"],  
columns=["Player_1 (prev)", "Player_2 (prev)"])
```

```
print(cnf_table)
```

```
# Gera a matriz de probabilidades da base completa
```

```
prb = LogisticRegression(random_state=0,max_iter=1000).fit(X, y)
```

```
prb.predict_proba(X)
```

```
# Salva a matriz de probabilidades em uma planilha Excel
```

```
df_odds = pd.DataFrame(prb.predict_proba(X))
```

```
df_odds.to_excel("C:\\Giba\\Dados\\_A_Giba\\Documentos\\Adm\\PUC_MG\\Python\\  
Bases_ML\\odds_predict_logistic_regression.xlsx")
```

```
# ## Gera o Modelo de Machine Learning utilizando o algoritmo de árvore de  
decisão
```

```
# As etapas de separação das colunas de dimensão e da coluna de classe de
classificação da base e
# separação das bases de treinamento e teste já foram executadas na criação do
modelo através do algoritmo de
# Logistic Regression
```

```
from sklearn.tree import DecisionTreeClassifier
```

```
# Executa o treinamento do modelo
```

```
#         odds_tree         =         DecisionTreeClassifier(random_state=0,
criterion='gini',min_samples_leaf=5,min_samples_split=5,max_depth=None)
odds_tree = DecisionTreeClassifier(random_state=0, criterion='entropy')
odds_tree = odds_tree.fit(x_train, y_train)
```

```
# Aplica o modelo na base de teste
```

```
y_pred = odds_tree.predict(x_test)
```

```
# Verifica a acurácia do modelo
```

```
score = accuracy_score(y_test, y_pred)
print(score)
```

```
# Verifica a performance de classificação do modelo
```

```
print(classification_report(y_test, y_pred, target_names=["Player_1", "Player_2"]))
```

```
# Verifica a matriz de confusão do modelo
```

```
cnf_matrix = confusion_matrix(y_test, y_pred)
cnf_table = pd.DataFrame(data=cnf_matrix, index=["Player_1", "Player_2"],
columns=["Player_1 (prev)", "Player_2 (prev)"])
print(cnf_table)
```

```
# Gera a matriz de probabilidades da base completa
```

```
prb = DecisionTreeClassifier(random_state=0, criterion='entropy').fit(X, y)
prb.predict_proba(X)
```

```
# ## Gera o Modelo de Machine Learning utilizando o algoritmo de Naive Bayes
```

```
# As etapas de separação das colunas de dimensão e da coluna de classe de
classificação da base e
```

```
# separação das bases de treinamento e teste já foram executadas na criação do
modelo através do algoritmo de
```

```
# Logistic Regression
```

```
from sklearn.naive_bayes import GaussianNB
```

```
# Executa o treinamento do modelo
```

```
gnb = GaussianNB()
gnb.fit(x_train, y_train)
```

```
# Aplica o modelo na base de teste
```

```
y_pred = gnb.predict(x_test)
```

```
# Verifica a acurácia do modelo
```

```
score = accuracy_score(y_test, y_pred)
```

```
print(score)
```

```
# Verifica a performance de classificação do modelo
```

```
print(classification_report(y_test, y_pred, target_names=["Player 1", "Player 2"]))
```

```
# Verifica a matriz de confusão do modelo
```

```
cnf_matrix = confusion_matrix(y_test, y_pred)
```

```
cnf_table = pd.DataFrame(data=cnf_matrix, index=["Player 1", "Player 2"],  
columns=["Player 1 (prev)", "Player 2 (prev)"])
```

```
print(cnf_table)
```

```
# Gera a matriz de probabilidades da base completa
```

```
prb = DecisionTreeClassifier(random_state=0, criterion='entropy').fit(X, y)
```

```
prb.predict_proba(X)
```

## **Tabelas**

### **DE-PARA de Torneios**

<b>Year</b>	<b>ATP</b>	<b>Location</b>	<b>Tournament</b>	<b>tourney_id</b>
2001	1	Adelaide	AAPT Championships	2001-339
2001	2	Chennai	TATA Open	2001-891
2001	3	Doha	Qatar Open	2001-451
2001	4	Auckland	Heineken Open	2001-301
2001	5	Sydney	adidas International	2001-338
2001	6	Melbourne	Australian Open	2001-580
2001	7	Bogota	Colombia Open	2001-359
2001	8	Milan	Milan Indoors	2001-408
2001	9	Copenhagen	Copenhagen Open	2001-481
2001	10	Marseille	Marseille Open	2001-496
2001	11	Vina del Mar	Bellsouth Open	2001-505
2001	12	Buenos Aires	Copa AT&T	2001-506
2001	13	Memphis	Kroger St. Jude	2001-402
2001	14	Rotterdam	ABN AMRO World Tennis Tournament	2001-407
2001	15	Acapulco	Abierto Mexicano	2001-807
2001	16	Dubai	Dubai Championships	2001-495
2001	17	San Jose	Sybase Open	2001-424
2001	18	Delray Beach	Citrix Tennis Championships	2001-499
2001	19	Scottsdale	Franklin Templeton Tennis Classic	2001-433
2001	20	Indian Wells	Indian Wells TMS	2001-404
2001	21	Miami	Ericsson Open	2001-403
2001	22	Casablanca	Grand Prix Hassan II	2001-360
2001	23	Estoril	Estoril Open	2001-468
2001	24	Monte Carlo	Monte Carlo Masters	2001-410
2001	25	Atlanta	Verizon Tennis Challenge	2001-409
2001	26	Barcelona	Open Seat Godo	2001-425
2001	27	Houston	U.S. Men's Clay Court Championships	2001-717
2001	28	Mallorca	Mallorca Open	2001-573
2001	29	Munich	BMW Open	2001-308
2001	30	Rome	Rome TMS	2001-416
2001	31	Hamburg	Hamburg TMS	2001-414
2001	32	St. Polten	Internationaler Raiffeisen Grand Prix	2001-306
2001	33	Paris	French Open	2001-520
2001	34	Halle	Gerry Weber Open	2001-500
2001	35	Queens Club	Stella Artois	2001-311
2001	36	Nottingham	Nottingham Open	2001-741
2001	37	's-Hertogenbosch	Heineken Trophy	2001-440
2001	38	London	Wimbledon	2001-540

2001	39	Bastad	Swedish Open	2001-316
2001	40	Gstaad	Gstaad Open	2001-314
2001	41	Newport	Hall of Fame Championships	2001-315
2001	42	Amsterdam	Dutch Open	2001-317
2001	43	Stuttgart	Mercedes Cup	2001-321
2001	44	Umag	Croatia Open	2001-439
2001	45	Kitzbuhel	Generali Open	2001-319
2001	46	Los Angeles	Mercedes-Benz Cup	2001-423
2001	47	Sopot	Idea Prokom Open	2001-475
2001	48	Montreal	Montreal TMS	2001-421
2001	49	Cincinnati	Cincinnati TMS	2001-422
2001	50	Indianapolis	RCA Championships	2001-419
2001	51	Washington	Legg Mason Classic	2001-418
2001	52	Long Island	Hamlet Cup	2001-441
2001	53	New York	US Open	2001-560
2001	54	Bucharest	Gelsor Open	2001-773
2001	55	Salvador	Brasil Open	2001-533
2001	56	Tashkent	President's Cup	2001-73
2001	57	Shanghai	Heineken Open	2001-890
2001	58	Hong Kong	Salem Open	2001-336
2001	59	Palermo	Campionati Internazionali Di Sicilia	2001-325
2001	60	Moscow	Kremlin Cup	2001-438
2001	61	Tokyo	Japan Open	2001-329
2001	62	Lyon	Grand Prix de Lyon	2001-375
2001	63	Vienna	CA Tennis Trophy	2001-337
2001	64	Stuttgart	Stuttgart TMS	2001-357
2001	65	Basel	Swiss Indoors	2001-328
2001	66	St. Petersburg	St. Petersburg Open	2001-568
2001	67	Stockholm	Stockholm Open	2001-429
2001	68	Paris	BNP Paribas	2001-352
2001	69	Sydney	Masters Cup	2001-605
2002	1	Adelaide	AAPT Championships	2002-339
2002	2	Chennai	TATA Open	2002-891
2002	3	Doha	Qatar Open	2002-451
2002	4	Auckland	Heineken Open	2002-301
2002	5	Sydney	adidas International	2002-338
2002	6	Melbourne	Australian Open	2002-580
2002	7	Milan	Milan Indoors	2002-408
2002	8	Copenhagen	Copenhagen Open	2002-481
2002	9	Marseille	Marseille Open	2002-496
2002	10	Vina del Mar	Bellsouth Open	2002-505
2002	11	Buenos Aires	Copa AT&T	2002-506
2002	12	Memphis	Kroger St. Jude	2002-402
2002	13	Rotterdam	ABN AMRO World Tennis Tournament	2002-407
2002	14	Acapulco	Abierto Mexicano	2002-807

2002	15	Dubai	Dubai Championships	2002-495
2002	16	San Jose	Sybase Open	2002-424
2002	17	Delray Beach	Citrix Tennis Championships	2002-499
2002	18	Scottsdale	Franklin Templeton Tennis Classic	2002-433
2002	19	Indian Wells	Pacific Life Open	2002-404
2002	20	Miami	NASDAQ-100 Open	2002-403
2002	21	Casablanca	Grand Prix Hassan II	2002-360
2002	22	Estoril	Estoril Open	2002-468
2002	23	Monte Carlo	Monte Carlo Masters	2002-410
2002	24	Barcelona	Open Seat Godo	2002-425
2002	25	Houston	U.S. Men's Clay Court Championships	2002-717
2002	26	Mallorca	Mallorca Open	2002-573
2002	27	Munich	BMW Open	2002-308
2002	28	Rome	Rome TMS	2002-416
2002	29	Hamburg	Hamburg TMS	2002-414
2002	30	St. Polten	Internationaler Raiffeisen Grand Prix	2002-306
2002	31	Paris	French Open	2002-520
2002	32	Halle	Gerry Weber Open	2002-500
2002	33	Queens Club	Queens Club	2002-311
2002	34	Nottingham	Nottingham Open	2002-741
2002	35	's-Hertogenbosch	Ordina Open	2002-440
2002	36	London	Wimbledon	2002-540
2002	37	Bastad	Swedish Open	2002-316
2002	38	Gstaad	Gstaad Open	2002-314
2002	39	Newport	Hall of Fame Championships	2002-315
2002	40	Amersfoort	Energis Open	2002-317
2002	41	Stuttgart	Mercedes Cup	2002-321
2002	42	Umag	Croatia Open	2002-439
2002	43	Kitzbuhel	Generali Open	2002-319
2002	44	Los Angeles	Mercedes-Benz Cup	2002-423
2002	45	Sopot	Idea Prokom Open	2002-475
2002	46	Toronto	Toronto TMS	2002-421
2002	47	Cincinnati	Cincinnati TMS	2002-422
2002	48	Indianapolis	RCA Championships	2002-419
2002	49	Washington	Legg Mason Classic	2002-418
2002	50	Long Island	TD Waterhouse Cup	2002-441
2002	51	New York	US Open	2002-560
2002	52	Bucharest	Open Romania	2002-773
2002	53	Salvador	Brasil Open	2002-533
2002	54	Tashkent	President's Cup	2002-73
2002	55	Hong Kong	Salem Open	2002-336
2002	56	Palermo	Campionati Internazionali Di Sicilia	2002-325
2002	57	Moscow	Kremlin Cup	2002-438
2002	58	Tokyo	Japan Open	2002-329
2002	59	Lyon	Grand Prix de Lyon	2002-375

2002	60	Vienna	CA Tennis Trophy	2002-337
2002	61	Madrid	Madrid Masters	2002-1536
2002	62	Basel	Swiss Indoors	2002-328
2002	63	St. Petersburg	St. Petersburg Open	2002-568
2002	64	Stockholm	Stockholm Open	2002-429
2002	65	Paris	BNP Paribas	2002-352
2002	66	Shanghai	Masters Cup	2002-605
2003	1	Adelaide	AAPT Championships	2003-339
2003	2	Chennai	TATA Open	2003-891
2003	3	Doha	Qatar Exxon Mobil Open	2003-451
2003	4	Auckland	Heineken Open	2003-301
2003	5	Sydney	adidas International	2003-338
2003	6	Melbourne	Australian Open	2003-580
2003	7	Milan	Breil ATP	2003-408
2003	8	Marseille	Open 13	2003-496
2003	9	San Jose	Siebel Open	2003-424
2003	10	Vina del Mar	Bellsouth Open	2003-505
2003	11	Buenos Aires	Copa AT&T	2003-506
2003	12	Memphis	Kroger St. Jude	2003-402
2003	13	Rotterdam	ABN AMRO World Tennis Tournament	2003-407
2003	14	Acapulco	Abierto Mexicano	2003-807
2003	15	Copenhagen	Copenhagen Open	2003-481
2003	16	Dubai	Dubai Championships	2003-495
2003	17	Delray Beach	International Championships	2003-499
2003	18	Scottsdale	Franklin Templeton Tennis Classic	2003-433
2003	19	Indian Wells	Pacific Life Open	2003-404
2003	20	Miami	NASDAQ-100 Open	2003-403
2003	21	Casablanca	Grand Prix Hassan II	2003-360
2003	22	Estoril	Estoril Open	2003-468
2003	23	Monte Carlo	Monte Carlo Masters	2003-410
2003	24	Barcelona	Open Seat Godo	2003-425
2003	25	Houston	U.S. Men's Clay Court Championships	2003-717
2003	26	Munich	BMW Open	2003-308
2003	27	Valencia	CAM Open Comunidad Valenciana	2003-573
2003	28	Rome	Telecom Italia Masters Roma	2003-416
2003	29	Hamburg	Hamburg TMS	2003-414
2003	30	St. Polten	Internationaler Raiffeisen Grand Prix	2003-306
2003	31	Paris	French Open	2003-520
2003	32	Halle	Gerry Weber Open	2003-500
2003	33	Queens Club	Stella Artois	2003-311
2003	34	Nottingham	Nottingham Open	2003-741
2003	35	's-Hertogenbosch	Ordina Open	2003-440
2003	36	London	Wimbledon	2003-540
2003	37	Bastad	Swedish Open	2003-316
2003	38	Gstaad	Allianz Suisse Open	2003-314



2003	39	Newport	Hall of Fame Championships	2003-315
2003	40	Amersfoort	Priority Telecom Dutch Open	2003-317
2003	41	Stuttgart	Mercedes Cup	2003-321
2003	42	Indianapolis	RCA Championships	2003-419
2003	43	Kitzbuhel	Generali Open	2003-319
2003	44	Umag	Croatia Open	2003-439
2003	45	Los Angeles	Mercedes-Benz Cup	2003-423
2003	46	Sopot	Idea Prokom Open	2003-475
2003	47	Washington	Legg Mason Classic	2003-418
2003	48	Montreal	Montreal TMS	2003-421
2003	49	Cincinnati	Western & Southern Financial Group Masters	2003-422
2003	50	Long Island	TD Waterhouse Cup	2003-441
2003	51	New York	US Open	2003-560
2003	52	Costa Do Sauipe	Brasil Open	2003-533
2003	53	Bucharest	Open Romania	2003-773
2003	54	Bangkok	Thailand Open	2003-1720
2003	55	Palermo	Campionati Internazionali Di Sicilia	2003-325
2003	56	Shanghai	Heineken Open	2003-890
2003	57	Metz	Open de Moselle	2003-341
2003	58	Moscow	Kremlin Cup	2003-438
2003	59	Tokyo	Japan Open	2003-329
2003	60	Lyon	Grand Prix de Lyon	2003-375
2003	61	Vienna	CA Tennis Trophy	2003-337
2003	62	Madrid	Madrid Masters	2003-1536
2003	63	Basel	Swiss Indoors	2003-328
2003	64	St. Petersburg	St. Petersburg Open	2003-568
2003	65	Stockholm	Stockholm Open	2003-429
2003	66	Paris	BNP Paribas	2003-352
2003	67	Houston	Masters Cup	2003-605
2004	1	Adelaide	AAPT Championships	2004-339
2004	2	Chennai	TATA Open	2004-891
2004	3	Doha	Qatar Exxon Mobil Open	2004-451
2004	4	Auckland	Heineken Open	2004-301
2004	5	Sydney	adidas International	2004-338
2004	6	Melbourne	Australian Open	2004-580
2004	7	Milan	Indesit ATP Milano Indoor	2004-408
2004	8	San Jose	Siebel Open	2004-424
2004	9	Vina del Mar	Bellsouth Open	2004-505
2004	10	Buenos Aires	ATP Buenos Aires 2004	2004-506
2004	11	Memphis	Kroger St. Jude	2004-402
2004	12	Rotterdam	ABN AMRO World Tennis Tournament	2004-407
2004	13	Costa Do Sauipe	Brasil Open	2004-533
2004	14	Marseille	Open 13	2004-496
2004	15	Acapulco	Abierto Mexicano	2004-807
2004	16	Dubai	Dubai Championships	2004-495

2004	17	Scottsdale	Franklin Templeton Tennis Classic	2004-433
2004	18	Indian Wells	Pacific Life Open	2004-404
2004	19	Miami	NASDAQ-100 Open	2004-403
2004	20	Estoril	Estoril Open	2004-468
2004	21	Houston	U.S. Men's Clay Court Championships	2004-717
2004	22	Valencia	CAM Open Comunidad Valenciana	2004-573
2004	23	Monte Carlo	Monte Carlo Masters	2004-410
2004	24	Barcelona	Open Seat Godo	2004-425
2004	25	Munich	BMW Open	2004-308
2004	26	Rome	Telecom Italia Masters Roma	2004-416
2004	27	Hamburg	Hamburg TMS	2004-414
2004	28	Casablanca	Grand Prix Hassan II	2004-360
2004	29	St. Polten	Internationaler Raiffeisen Grand Prix	2004-306
2004	30	Paris	French Open	2004-520
2004	31	Halle	Gerry Weber Open	2004-500
2004	32	Queens Club	Stella Artois	2004-311
2004	33	's-Hertogenbosch	Ordina Open	2004-440
2004	34	Nottingham	Nottingham Open	2004-741
2004	35	London	Wimbledon	2004-540
2004	36	Bastad	Swedish Open	2004-316
2004	37	Gstaad	Allianz Suisse Open	2004-314
2004	38	Newport	Hall of Fame Championships	2004-315
2004	39	Amersfoort	Priority Telecom Dutch Open	2004-317
2004	40	Los Angeles	Mercedes-Benz Cup	2004-423
2004	41	Stuttgart	Mercedes Cup	2004-321
2004	42	Indianapolis	RCA Championships	2004-419
2004	43	Kitzbuhel	Generali Open	2004-319
2004	44	Umag	Croatia Open	2004-439
2004	45	Toronto	Toronto TMS	2004-421
2004	46	Cincinnati	Western & Southern Financial Group Masters	2004-422
2004	47	Sopot	Idea Prokom Open	2004-475
2004	48	Washington	Legg Mason Classic	2004-418
2004	49	Long Island	TD Waterhouse Cup	2004-441
2004	50	New York	US Open	2004-560
2004	51	Beijing	China Open	2004-747
2004	52	Bucharest	Open Romania	2004-773
2004	53	Delray Beach	International Championships	2004-499
2004	54	Bangkok	Thailand Open	2004-1720
2004	55	Palermo	Campionati Internazionali Di Sicilia	2004-325
2004	56	Shanghai	Heineken Open	2004-890
2004	57	Lyon	Grand Prix de Lyon	2004-375
2004	58	Tokyo	Japan Open	2004-329
2004	59	Metz	Open de Moselle	2004-341
2004	60	Moscow	Kremlin Cup	2004-438
2004	61	Vienna	CA Tennis Trophy	2004-337

2004	62	Madrid	Madrid Masters	2004-1536
2004	63	Basel	Swiss Indoors	2004-328
2004	64	St. Petersburg	St. Petersburg Open	2004-568
2004	65	Stockholm	Stockholm Open	2004-429
2004	66	Paris	BNP Paribas	2004-352
2004	67	Houston	Masters Cup	2004-605
2005	1	Adelaide	Next Generation Hardcourts	2005-339
2005	2	Chennai	TATA Open	2005-891
2005	3	Doha	Qatar Exxon Mobil Open	2005-451
2005	4	Auckland	Heineken Open	2005-301
2005	5	Sydney	Medibank International	2005-338
2005	6	Melbourne	Australian Open	2005-580
2005	7	Delray Beach	International Championships	2005-499
2005	8	Milan	Internazionali di Lombardia	2005-408
2005	9	Vina del Mar	Bellsouth Open	2005-505
2005	10	Buenos Aires	ATP Buenos Aires 2005	2005-506
2005	11	Marseille	Open 13	2005-496
2005	12	San Jose	SAP Open	2005-424
2005	13	Costa Do Sauipe	Brasil Open	2005-533
2005	14	Memphis	Regions Morgan Keegan Championships	2005-402
2005	15	Rotterdam	ABN AMRO World Tennis Tournament	2005-407
2005	16	Acapulco	Abierto Mexicano	2005-807
2005	17	Dubai	Dubai Duty Free Men's Open	2005-495
2005	18	Scottsdale	Channel Open	2005-433
2005	19	Indian Wells	Pacific Life Open	2005-404
2005	20	Miami	NASDAQ-100 Open	2005-403
2005	21	Casablanca	Grand Prix Hassan II	2005-360
2005	22	Valencia	Open de Tennis Comunidad Valenciana	2005-573
2005	23	Monte Carlo	Monte Carlo Masters	2005-410
2005	24	Barcelona	Open Seat Godo	2005-425
2005	25	Houston	U.S. Men's Clay Court Championships	2005-717
2005	26	Estoril	Estoril Open	2005-468
2005	27	Munich	BMW Open	2005-308
2005	28	Rome	Telecom Italia Masters Roma	2005-416
2005	29	Hamburg	Hamburg TMS	2005-414
2005	30	St. Polten	Internationaler Raiffeisen Grand Prix	2005-306
2005	31	Paris	French Open	2005-520
2005	32	Halle	Gerry Weber Open	2005-500
2005	33	Queens Club	Stella Artois	2005-311
2005	34	's-Hertogenbosch	Ordina Open	2005-440
2005	35	Nottingham	Nottingham Open	2005-741
2005	36	London	Wimbledon	2005-540
2005	37	Bastad	Swedish Open	2005-316
2005	38	Gstaad	Allianz Suisse Open	2005-314
2005	39	Newport	Hall of Fame Championships	2005-315

2005	40	Amersfoort	Priority Telecom Dutch Open	2005-317
2005	41	Indianapolis	RCA Championships	2005-419
2005	42	Stuttgart	Mercedes Cup	2005-321
2005	43	Kitzbuhel	Generali Open	2005-319
2005	44	Los Angeles	Mercedes-Benz Cup	2005-423
2005	45	Umag	Croatia Open	2005-439
2005	46	Sopot	Idea Prokom Open	2005-475
2005	47	Washington	Legg Mason Classic	2005-418
2005	48	Montreal	Rogers Cup	2005-421
2005	49	Cincinnati	Western & Southern Financial Group Masters	2005-422
2005	50	New Haven	Pilot Pen Tennis	2005-3348
2005	51	New York	US Open	2005-560
2005	52	Beijing	China Open	2005-747
2005	53	Bucharest	Open Romania	2005-773
2005	54	Bangkok	Thailand Open	2005-1720
2005	55	Ho Chi Min City	Vietnam Open	2005-890
2005	56	Palermo	Campionati Internazionali Di Sicilia	2005-325
2005	57	Metz	Open de Moselle	2005-341
2005	58	Tokyo	Japan Open	2005-329
2005	59	Moscow	Kremlin Cup	2005-438
2005	60	Stockholm	Stockholm Open	2005-429
2005	61	Vienna	BA-CA Tennis Trophy	2005-337
2005	62	Madrid	Madrid Masters	2005-1536
2005	63	Basel	Davidoff Swiss Indoors	2005-328
2005	64	Lyon	Grand Prix de Lyon	2005-375
2005	65	St. Petersburg	St. Petersburg Open	2005-568
2005	66	Paris	BNP Paribas	2005-352
2005	67	Shanghai	Masters Cup	2005-605
2006	1	Adelaide	Next Generation Adelaide International	2006-339
2006	2	Chennai	Chennai Open	2006-891
2006	3	Doha	Qatar Exxon Mobil Open	2006-451
2006	4	Auckland	Heineken Open	2006-301
2006	5	Sydney	Sydney International	2006-338
2006	6	Melbourne	Australian Open	2006-580
2006	7	Delray Beach	International Championships	2006-499
2006	8	Vina del Mar	Movistar Open	2006-505
2006	9	Zagreb	PBZ Zagreb Indoors	2006-2276
2006	10	Buenos Aires	Copa Telmex	2006-506
2006	11	Marseille	Open 13	2006-496
2006	12	San Jose	SAP Open	2006-424
2006	13	Costa Do Sauipe	Brasil Open	2006-533
2006	14	Memphis	Regions Morgan Keegan Championships	2006-402
2006	15	Rotterdam	ABN AMRO World Tennis Tournament	2006-407
2006	16	Acapulco	Abierto Mexicano	2006-807
2006	17	Dubai	Dubai Tennis Championships	2006-495

2006	18	Las Vegas	Channel Open	2006-433
2006	19	Indian Wells	Pacific Life Open	2006-404
2006	20	Miami	NASDAQ-100 Open	2006-403
2006	21	Houston	U.S. Men's Clay Court Championships	2006-717
2006	22	Valencia	Open de Tennis Comunidad Valenciana	2006-573
2006	23	Monte Carlo	Monte Carlo Masters	2006-410
2006	24	Barcelona	Open Seat Godo	2006-425
2006	25	Casablanca	Grand Prix Hassan II	2006-360
2006	26	Estoril	Estoril Open	2006-468
2006	27	Munich	BMW Open	2006-308
2006	28	Rome	Campionati Internazionali d'Italia	2006-416
2006	29	Hamburg	Hamburg TMS	2006-414
2006	30	Portschach	Hypo Group Tennis International	2006-306
2006	31	Paris	French Open	2006-520
2006	32	Halle	Gerry Weber Open	2006-500
2006	33	Queens Club	Stella Artois	2006-311
2006	34	's-Hertogenbosch	Ordina Open	2006-440
2006	35	Nottingham	Red Letter Days Open	2006-741
2006	36	London	Wimbledon	2006-540
2006	37	Bastad	Synsam Swedish Open	2006-316
2006	38	Gstaad	Allianz Suisse Open	2006-314
2006	39	Newport	Hall of Fame Championships	2006-315
2006	40	Amersfoort	Dutch Open	2006-317
2006	41	Indianapolis	RCA Championships	2006-419
2006	42	Stuttgart	Mercedes Cup	2006-321
2006	43	Kitzbuhel	Generali Open	2006-319
2006	44	Los Angeles	Countrywide Classic	2006-423
2006	45	Umag	Croatia Open	2006-439
2006	46	Sopot	Idea Prokom Open	2006-475
2006	47	Washington	Legg Mason Classic	2006-418
2006	48	Toronto	Rogers Masters	2006-421
2006	49	Cincinnati	Western & Southern Financial Group Masters	2006-422
2006	50	New Haven	Pilot Pen Tennis	2006-3348
2006	51	New York	US Open	2006-560
2006	52	Beijing	China Open	2006-747
2006	53	Bucharest	Open Romania	2006-773
2006	54	Bangkok	Thailand Open	2006-1720
2006	55	Mumbai	Kingfisher Airlines Tennis Open	2006-3465
2006	56	Palermo	Campionati Internazionali Di Sicilia	2006-325
2006	57	Metz	Open de Moselle	2006-341
2006	58	Tokyo	Japan Open	2006-329
2006	59	Moscow	Kremlin Cup	2006-438
2006	60	Stockholm	Stockholm Open	2006-429
2006	61	Vienna	BA-CA Tennis Trophy	2006-337
2006	62	Madrid	Madrid Masters	2006-1536

2006	63	Basel	Davidoff Swiss Indoors	2006-328
2006	64	Lyon	Grand Prix de Lyon	2006-375
2006	65	St. Petersburg	St. Petersburg Open	2006-568
2006	66	Paris	BNP Paribas	2006-352
2006	67	Shanghai	Masters Cup	2006-605
2007	1	Adelaide	Next Generation Adelaide International	2007-339
2007	2	Chennai	Chennai Open	2007-891
2007	3	Doha	Qatar Exxon Mobil Open	2007-451
2007	4	Auckland	Heineken Open	2007-301
2007	5	Sydney	Sydney International	2007-338
2007	6	Melbourne	Australian Open	2007-580
2007	7	Delray Beach	International Championships	2007-499
2007	8	Vina del Mar	Movistar Open	2007-505
2007	9	Zagreb	PBZ Zagreb Indoors	2007-2276
2007	10	Costa Do Sauipe	Brasil Open	2007-533
2007	11	Marseille	Open 13	2007-496
2007	12	San Jose	SAP Open	2007-424
2007	13	Buenos Aires	Copa Telmex	2007-506
2007	14	Memphis	Regions Morgan Keegan Championships	2007-402
2007	15	Rotterdam	ABN AMRO World Tennis Tournament	2007-407
2007	16	Acapulco	Abierto Mexicano	2007-807
2007	17	Dubai	Dubai Tennis Championships	2007-495
2007	18	Las Vegas	Channel Open	2007-433
2007	19	Indian Wells	Pacific Life Open	2007-404
2007	20	Miami	Sony Ericsson Open	2007-403
2007	21	Houston	U.S. Men's Clay Court Championships	2007-717
2007	22	Valencia	Open de Tenis Comunidad Valenciana	2007-573
2007	23	Monte Carlo	Monte Carlo Masters	2007-410
2007	24	Barcelona	Open Seat Godo	2007-425
2007	25	Casablanca	Grand Prix Hassan II	2007-360
2007	26	Estoril	Estoril Open	2007-468
2007	27	Munich	BMW Open	2007-308
2007	28	Rome	Internazionali BNL d'Italia	2007-416
2007	29	Hamburg	Hamburg TMS	2007-414
2007	30	Portschach	Hypo Group Tennis International	2007-306
2007	31	Paris	French Open	2007-520
2007	32	Halle	Gerry Weber Open	2007-500
2007	33	Queens Club	Stella Artois	2007-311
2007	34	's-Hertogenbosch	Ordina Open	2007-440
2007	35	Nottingham	The Nottingham Open	2007-741
2007	36	London	Wimbledon	2007-540
2007	37	Bastad	Catella Swedish Open	2007-316
2007	38	Gstaad	Allianz Suisse Open	2007-314
2007	39	Newport	Hall of Fame Championships	2007-315
2007	40	Amersfoort	Dutch Open	2007-317

2007	41	Los Angeles	Countrywide Classic	2007-423
2007	42	Stuttgart	Mercedes Cup	2007-321
2007	43	Indianapolis	Indianapolis Tennis Championships	2007-419
2007	44	Kitzbuhel	Austrian Open	2007-319
2007	45	Umag	Croatia Open	2007-439
2007	46	Sopot	Idea Prokom Open	2007-475
2007	47	Washington	Legg Mason Classic	2007-418
2007	48	Montreal	Rogers Masters	2007-421
2007	49	Cincinnati	Western & Southern Financial Group Masters	2007-422
2007	50	New Haven	Pilot Pen Tennis	2007-3348
2007	51	New York	US Open	2007-560
2007	52	Beijing	China Open	2007-747
2007	53	Bucharest	Open Romania	2007-773
2007	54	Bangkok	Thailand Open	2007-1720
2007	55	Mumbai	Kingfisher Airlines Tennis Open	2007-3465
2007	56	Metz	Open de Moselle	2007-341
2007	57	Tokyo	AIG Japan Open Tennis Championships	2007-329
2007	58	Moscow	Kremlin Cup	2007-438
2007	59	Stockholm	Stockholm Open	2007-429
2007	60	Vienna	BA-CA Tennis Trophy	2007-337
2007	61	Madrid	Madrid Masters	2007-1536
2007	62	Basel	Davidoff Swiss Indoors	2007-328
2007	63	Lyon	Grand Prix de Lyon	2007-375
2007	64	St. Petersburg	St. Petersburg Open	2007-568
2007	65	Paris	BNP Paribas	2007-352
2007	66	Shanghai	Masters Cup	2007-605
2008	1	Adelaide	Next Generation Adelaide International	2008-339
2008	2	Chennai	Chennai Open	2008-891
2008	3	Doha	Qatar Exxon Mobil Open	2008-451
2008	4	Auckland	Heineken Open	2008-301
2008	5	Sydney	Medibank International	2008-338
2008	6	Melbourne	Australian Open	2008-580
2008	7	Vina del Mar	Movistar Open	2008-505
2008	8	Costa Do Sauipe	Brasil Open	2008-533
2008	9	Delray Beach	International Championships	2008-499
2008	10	Marseille	Open 13	2008-496
2008	11	Buenos Aires	Copa Telmex	2008-506
2008	12	Rotterdam	ABN AMRO World Tennis Tournament	2008-407
2008	13	San Jose	SAP Open	2008-424
2008	14	Acapulco	Abierto Mexicano	2008-807
2008	15	Memphis	Regions Morgan Keegan Championships	2008-402
2008	16	Zagreb	PBZ Zagreb Indoors	2008-2276
2008	17	Dubai	Dubai Tennis Championships	2008-495
2008	18	Las Vegas	Channel Open	2008-433
2008	19	Indian Wells	Pacific Life Open	2008-404

2008	20	Miami	Sony Ericsson Open	2008-403
2008	21	Estoril	Estoril Open	2008-468
2008	22	Houston	U.S. Men's Clay Court Championships	2008-717
2008	23	Valencia	Open de Tennis Comunidad Valenciana	2008-573
2008	24	Monte Carlo	Monte Carlo Masters	2008-410
2008	25	Barcelona	Open Sabadell Atlántico 2008	2008-425
2008	26	Munich	BMW Open	2008-308
2008	27	Rome	Internazionali BNL d'Italia	2008-416
2008	28	Hamburg	Hamburg TMS	2008-414
2008	29	Casablanca	Grand Prix Hassan II	2008-360
2008	30	Portschach	Hypo Group Tennis International	2008-306
2008	31	Paris	French Open	2008-520
2008	32	Halle	Gerry Weber Open	2008-500
2008	33	Queens Club	Stella Artois	2008-311
2008	34	Warsaw	Orange Prokom Open	2008-475
2008	35	's-Hertogenbosch	Ordina Open	2008-440
2008	36	Nottingham	Slazenger Open	2008-741
2008	37	London	Wimbledon	2008-540
2008	38	Bastad	Catella Swedish Open	2008-316
2008	39	Gstaad	Allianz Suisse Open	2008-314
2008	40	Newport	Hall of Fame Championships	2008-315
2008	41	Stuttgart	Mercedes Cup	2008-321
2008	42	Amersfoort	Dutch Open	2008-317
2008	43	Indianapolis	Indianapolis Tennis Championships	2008-419
2008	44	Kitzbuhel	Austrian Open	2008-319
2008	45	Umag	Studena Croatia Open	2008-439
2008	46	Toronto	Rogers Masters	2008-421
2008	47	Cincinnati	Western & Southern Financial Group Masters	2008-422
2008	48	Los Angeles	Countrywide Classic	2008-423
2008	49	Washington	Legg Mason Classic	2008-418
2008	50	New Haven	Pilot Pen Tennis	2008-3348
2008	51	New York	US Open	2008-560
2008	52	Bucharest	Open Romania	2008-773
2008	53	Bangkok	Thailand Open	2008-1720
2008	54	Beijing	China Open	2008-747
2008	55	Metz	Open de Moselle	2008-341
2008	56	Tokyo	AIG Japan Open Tennis Championships	2008-329
2008	57	Moscow	Kremlin Cup	2008-438
2008	58	Stockholm	Stockholm Open	2008-429
2008	59	Vienna	BA-CA Tennis Trophy	2008-337
2008	60	Madrid	Madrid Masters	2008-1536
2008	61	Basel	Davidoff Swiss Indoors	2008-328
2008	62	Lyon	Grand Prix de Lyon	2008-375
2008	63	St. Petersburg	St. Petersburg Open	2008-568
2008	64	Paris	BNP Paribas	2008-352



2008	65	Shanghai	Masters Cup	2008-605
2009	1	Brisbane	Brisbane International	2009-339
2009	2	Chennai	Chennai Open	2009-891
2009	3	Doha	Qatar Exxon Mobil Open	2009-451
2009	4	Auckland	Heineken Open	2009-301
2009	5	Sydney	Medibank International	2009-338
2009	6	Melbourne	Australian Open	2009-580
2009	7	Johannesburg	SA Tennis Open	2009-5012
2009	8	Vina del Mar	Movistar Open	2009-505
2009	9	Zagreb	PBZ Zagreb Indoors	2009-2276
2009	10	Costa Do Sauipe	Brasil Open	2009-533
2009	11	Rotterdam	ABN AMRO World Tennis Tournament	2009-407
2009	12	San Jose	SAP Open	2009-424
2009	13	Buenos Aires	Copa Telmex	2009-506
2009	14	Marseille	Open 13	2009-496
2009	15	Memphis	Regions Morgan Keegan Championships	2009-402
2009	16	Acapulco	Abierto Mexicano	2009-807
2009	17	Delray Beach	International Championships	2009-499
2009	18	Dubai	Dubai Tennis Championships	2009-495
2009	19	Indian Wells	BNP Paribas Open	2009-404
2009	20	Miami	Sony Ericsson Open	2009-403
2009	21	Casablanca	Grand Prix Hassan II	2009-360
2009	22	Houston	U.S. Men's Clay Court Championships	2009-717
2009	23	Monte Carlo	Monte Carlo Masters	2009-410
2009	24	Barcelona	Open Banco Sabadell	2009-425
2009	25	Rome	Internazionali BNL d'Italia	2009-416
2009	26	Belgrade	Serbia Open	2009-5053
2009	27	Estoril	Estoril Open	2009-468
2009	28	Munich	BMW Open	2009-308
2009	29	Madrid	Mutua Madrileña Madrid Open	2009-1536
2009	30	Kitzbuhel	Austrian Open	2009-306
2009	31	Paris	French Open	2009-520
2009	32	Halle	Gerry Weber Open	2009-500
2009	33	Queens Club	AEGON Championships	2009-311
2009	34	Eastbourne	AEGON International	2009-741
2009	35	's-Hertogenbosch	Ordina Open	2009-440
2009	36	London	Wimbledon	2009-540
2009	37	Newport	Hall of Fame Championships	2009-315
2009	38	Bastad	Catella Swedish Open	2009-316
2009	39	Stuttgart	Mercedes Cup	2009-321
2009	40	Hamburg	International German Open	2009-414
2009	41	Indianapolis	Indianapolis Tennis Championships	2009-419
2009	42	Gstaad	Allianz Suisse Open	2009-314
2009	43	Los Angeles	LA Tennis Open	2009-423
2009	44	Umag	Studena Croatia Open	2009-439

2009	45	Washington	Legg Mason Classic	2009-418
2009	46	Montreal	Rogers Masters	2009-421
2009	47	Cincinnati	Western & Southern Financial Group Masters	2009-422
2009	48	New Haven	Pilot Pen Tennis	2009-3348
2009	49	New York	US Open	2009-560
2009	50	Bucharest	Open Romania	2009-773
2009	51	Metz	Open de Moselle	2009-341
2009	52	Bangkok	Thailand Open	2009-1720
2009	53	Kuala Lumpur	Proton Malaysian Open	2009-6003
2009	54	Beijing	China Open	2009-747
2009	55	Tokyo	Rakuten Japan Open Tennis Championships	2009-329
2009	56	Shanghai	Shanghai Masters	2009-5014
2009	57	Moscow	Kremlin Cup	2009-438
2009	58	Stockholm	Stockholm Open	2009-429
2009	59	Lyon	Grand Prix de Lyon	2009-375
2009	60	St. Petersburg	St. Petersburg Open	2009-568
2009	61	Vienna	BA-CA Tennis Trophy	2009-337
2009	62	Basel	Davidoff Swiss Indoors	2009-328
2009	63	Valencia	Valencia Open 500	2009-573
2009	64	Paris	BNP Paribas Masters	2009-352
2009	65	London	Masters Cup	2009-605
2010	1	Brisbane	Brisbane International	2010-339
2010	2	Chennai	Chennai Open	2010-891
2010	3	Doha	Qatar Exxon Mobil Open	2010-451
2010	4	Auckland	Heineken Open	2010-301
2010	5	Sydney	Medibank International	2010-338
2010	6	Melbourne	Australian Open	2010-580
2010	7	Johannesburg	SA Tennis Open	2010-5012
2010	8	Santiago	Movistar Open	2010-505
2010	9	Zagreb	PBZ Zagreb Indoors	2010-2276
2010	10	Costa Do Sauipe	Brasil Open	2010-533
2010	11	Rotterdam	ABN AMRO World Tennis Tournament	2010-407
2010	12	San Jose	SAP Open	2010-424
2010	13	Buenos Aires	Copa Telmex	2010-506
2010	14	Marseille	Open 13	2010-496
2010	15	Memphis	Regions Morgan Keegan Championships	2010-402
2010	16	Acapulco	Abierto Mexicano	2010-807
2010	17	Delray Beach	International Championships	2010-499
2010	18	Dubai	Dubai Tennis Championships	2010-495
2010	19	Indian Wells	BNP Paribas Open	2010-404
2010	20	Miami	Sony Ericsson Open	2010-403
2010	21	Casablanca	Grand Prix Hassan II	2010-360
2010	22	Houston	U.S. Men's Clay Court Championships	2010-717
2010	23	Monte Carlo	Monte Carlo Masters	2010-410
2010	24	Barcelona	Open Banco Sabadell	2010-425

2010	25	Rome	Internazionali BNL d'Italia	2010-416
2010	26	Belgrade	Serbia Open	2010-5053
2010	27	Estoril	Estoril Open	2010-468
2010	28	Munich	BMW Open	2010-308
2010	29	Madrid	Mutua Madrileña Madrid Open	2010-1536
2010	30	Nice	Open de Nice Côte d'Azur	2010-6120
2010	31	Paris	French Open	2010-520
2010	32	Halle	Gerry Weber Open	2010-500
2010	33	Queens Club	AEGON Championships	2010-311
2010	34	Eastbourne	AEGON International	2010-741
2010	35	's-Hertogenbosch	Unicef Open	2010-440
2010	36	London	Wimbledon	2010-540
2010	37	Newport	Hall of Fame Championships	2010-315
2010	38	Bastad	SkiStar Swedish Open	2010-316
2010	39	Stuttgart	Mercedes Cup	2010-321
2010	40	Atlanta	Atlanta Tennis Championships	2010-6116
2010	41	Hamburg	German Open Tennis Championships	2010-414
2010	42	Gstaad	Allianz Suisse Open	2010-314
2010	43	Los Angeles	Farmers Classic	2010-423
2010	44	Umag	Studena Croatia Open	2010-439
2010	45	Washington	Legg Mason Classic	2010-418
2010	46	Toronto	Rogers Masters	2010-421
2010	47	Cincinnati	Western & Southern Financial Group Masters	2010-422
2010	48	New Haven	Pilot Pen Tennis	2010-3348
2010	49	New York	US Open	2010-560
2010	50	Bucharest	Open Romania	2010-773
2010	51	Metz	Open de Moselle	2010-341
2010	52	Bangkok	Thailand Open	2010-1720
2010	53	Kuala Lumpur	Proton Malaysian Open	2010-6003
2010	54	Beijing	China Open	2010-747
2010	55	Tokyo	Rakuten Japan Open Tennis Championships	2010-329
2010	56	Shanghai	Shanghai Masters	2010-5014
2010	57	Moscow	Kremlin Cup	2010-438
2010	58	Stockholm	Stockholm Open	2010-429
2010	59	Montpellier	Open Sud de France	2010-375
2010	60	St. Petersburg	St. Petersburg Open	2010-568
2010	61	Vienna	BA-CA Tennis Trophy	2010-337
2010	62	Basel	Swiss Indoors	2010-328
2010	63	Valencia	Valencia Open 500	2010-573
2010	64	Paris	BNP Paribas Masters	2010-352
2010	65	London	Masters Cup	2010-605
2011	1	Brisbane	Brisbane International	2011-339
2011	2	Chennai	Chennai Open	2011-891
2011	3	Doha	Qatar Exxon Mobil Open	2011-451
2011	4	Auckland	Heineken Open	2011-301

2011	5	Sydney	Medibank International	2011-338
2011	6	Melbourne	Australian Open	2011-580
2011	7	Johannesburg	SA Tennis Open	2011-5012
2011	8	Santiago	Movistar Open	2011-505
2011	9	Zagreb	PBZ Zagreb Indoors	2011-2276
2011	10	Costa Do Sauipe	Brasil Open	2011-533
2011	11	Rotterdam	ABN AMRO World Tennis Tournament	2011-407
2011	12	San Jose	SAP Open	2011-424
2011	13	Buenos Aires	Copa Claro	2011-506
2011	14	Marseille	Open 13	2011-496
2011	15	Memphis	Regions Morgan Keegan Championships	2011-402
2011	16	Acapulco	Abierto Mexicano	2011-807
2011	17	Delray Beach	International Championships	2011-499
2011	18	Dubai	Dubai Tennis Championships	2011-495
2011	19	Indian Wells	BNP Paribas Open	2011-404
2011	20	Miami	Sony Ericsson Open	2011-403
2011	21	Casablanca	Grand Prix Hassan II	2011-360
2011	22	Houston	U.S. Men's Clay Court Championships	2011-717
2011	23	Monte Carlo	Monte Carlo Masters	2011-410
2011	24	Barcelona	Open Banco Sabadell	2011-425
2011	25	Belgrade	Serbia Open	2011-5053
2011	26	Estoril	Estoril Open	2011-468
2011	27	Munich	BMW Open	2011-308
2011	28	Madrid	Mutua Madrileña Madrid Open	2011-1536
2011	29	Rome	Internazionali BNL d'Italia	2011-416
2011	30	Nice	Open de Nice Côte d'Azur	2011-6120
2011	31	Paris	French Open	2011-520
2011	32	Halle	Gerry Weber Open	2011-500
2011	33	Queens Club	AEGON Championships	2011-311
2011	34	Eastbourne	AEGON International	2011-741
2011	35	's-Hertogenbosch	Unicef Open	2011-440
2011	36	London	Wimbledon	2011-540
2011	37	Newport	Hall of Fame Championships	2011-315
2011	38	Bastad	SkiStar Swedish Open	2011-316
2011	39	Stuttgart	Mercedes Cup	2011-321
2011	40	Atlanta	Atlanta Tennis Championships	2011-6116
2011	41	Hamburg	German Open Tennis Championships	2011-414
2011	42	Gstaad	Allianz Suisse Open	2011-314
2011	43	Los Angeles	Farmers Classic	2011-423
2011	44	Umag	Studena Croatia Open	2011-439
2011	45	Kitzbuhel	Bet-At-Home Cup	2011-319
2011	46	Washington	Legg Mason Classic	2011-418
2011	47	Montreal	Rogers Masters	2011-421
2011	48	Cincinnati	Western & Southern Financial Group Masters	2011-422
2011	49	Winston-Salem	Winston-Salem Open at Wake Forest University	2011-6242

2011	50	New York	US Open	2011-560
2011	51	Bucharest	Open Romania	2011-773
2011	52	Metz	Open de Moselle	2011-341
2011	53	Bangkok	Thailand Open	2011-1720
2011	54	Kuala Lumpur	Malaysian Open	2011-6003
2011	55	Beijing	China Open	2011-747
2011	56	Tokyo	Rakuten Japan Open Tennis Championships	2011-329
2011	57	Shanghai	Shanghai Masters	2011-5014
2011	58	Moscow	Kremlin Cup	2011-438
2011	59	Stockholm	Stockholm Open	2011-429
2011	60	St. Petersburg	St. Petersburg Open	2011-568
2011	61	Vienna	Erste Bank Open	2011-337
2011	62	Basel	Swiss Indoors	2011-328
2011	63	Valencia	Valencia Open 500	2011-573
2011	64	Paris	BNP Paribas Masters	2011-352
2011	65	London	Masters Cup	2011-605
2012	1	Brisbane	Brisbane International	2012-339
2012	2	Chennai	Chennai Open	2012-891
2012	3	Doha	Qatar Exxon Mobil Open	2012-451
2012	4	Auckland	Heineken Open	2012-301
2012	5	Sydney	Apia International	2012-338
2012	6	Melbourne	Australian Open	2012-580
2012	7	Montpellier	Open Sud de France	2012-375
2012	8	Vina del Mar	VTR Open	2012-505
2012	9	Zagreb	PBZ Zagreb Indoors	2012-2276
2012	10	Rotterdam	ABN AMRO World Tennis Tournament	2012-407
2012	11	San Jose	SAP Open	2012-424
2012	12	Sao Paulo	Brasil Open	2012-533
2012	13	Buenos Aires	Copa Claro	2012-506
2012	14	Marseille	Open 13	2012-496
2012	15	Memphis	Regions Morgan Keegan Championships	2012-402
2012	16	Acapulco	Abierto Mexicano	2012-807
2012	17	Delray Beach	International Championships	2012-499
2012	18	Dubai	Dubai Tennis Championships	2012-495
2012	19	Indian Wells	BNP Paribas Open	2012-404
2012	20	Miami	Sony Ericsson Open	2012-403
2012	21	Casablanca	Grand Prix Hassan II	2012-360
2012	22	Houston	U.S. Men's Clay Court Championships	2012-717
2012	23	Monte Carlo	Monte Carlo Masters	2012-410
2012	24	Barcelona	Open Banco Sabadell	2012-425
2012	25	Bucharest	BRD Nastase Tiriach Trophy	2012-773
2012	26	Belgrade	Serbia Open	2012-5053
2012	27	Estoril	Estoril Open	2012-468
2012	28	Munich	BMW Open	2012-308
2012	29	Madrid	Mutua Madrid Open	2012-1536

2012	30	Rome	Internazionali BNL d'Italia	2012-416
2012	31	Nice	Open de Nice Côte d'Azur	2012-6120
2012	32	Paris	French Open	2012-520
2012	33	Halle	Gerry Weber Open	2012-500
2012	34	Queens Club	AEGON Championships	2012-311
2012	35	Eastbourne	AEGON International	2012-741
2012	36	's-Hertogenbosch	Unicef Open	2012-440
2012	37	London	Wimbledon	2012-540
2012	38	Bastad	SkiStar Swedish Open	2012-316
2012	39	Newport	Hall of Fame Championships	2012-315
2012	40	Stuttgart	Mercedes Cup	2012-321
2012	41	Umag	ATP Vegeta Croatia Open	2012-439
2012	42	Atlanta	BB&T Atlanta Open	2012-6116
2012	43	Gstaad	Crédit Agricole Suisse Open Gstaad	2012-314
2012	44	Hamburg	German Open Tennis Championships	2012-414
2012	45	Kitzbuhel	Bet-At-Home Cup	2012-319
2012	46	Los Angeles	Farmers Classic	2012-423
2012	47	Washington	Citi Open	2012-418
2012	48	Toronto	Rogers Masters	2012-421
2012	49	Cincinnati	Western & Southern Financial Group Masters	2012-422
2012	50	Winston-Salem	Winston-Salem Open at Wake Forest University	2012-6242
2012	51	New York	US Open	2012-560
2012	52	Metz	Open de Moselle	2012-341
2012	53	St. Petersburg	St. Petersburg Open	2012-568
2012	54	Bangkok	Thailand Open	2012-1720
2012	55	Kuala Lumpur	Malaysian Open	2012-6003
2012	56	Beijing	China Open	2012-747
2012	57	Tokyo	Rakuten Japan Open Tennis Championships	2012-329
2012	58	Shanghai	Shanghai Masters	2012-5014
2012	59	Moscow	Kremlin Cup	2012-438
2012	60	Stockholm	Stockholm Open	2012-429
2012	61	Vienna	Erste Bank Open	2012-337
2012	62	Basel	Swiss Indoors	2012-328
2012	63	Valencia	Valencia Open 500	2012-573
2012	64	Paris	BNP Paribas Masters	2012-352
2012	65	London	Masters Cup	2012-605
2013	1	Brisbane	Brisbane International	2013-339
2013	2	Chennai	Chennai Open	2013-891
2013	3	Doha	Qatar Exxon Mobil Open	2013-451
2013	4	Auckland	Heineken Open	2013-301
2013	5	Sydney	Apia International	2013-338
2013	6	Melbourne	Australian Open	2013-580
2013	7	Montpellier	Open Sud de France	2013-375
2013	8	Vina del Mar	VTR Open	2013-505
2013	9	Zagreb	PBZ Zagreb Indoors	2013-2276

2013	10	Rotterdam	ABN AMRO World Tennis Tournament	2013-407
2013	11	San Jose	SAP Open	2013-424
2013	12	Sao Paulo	Brasil Open	2013-533
2013	13	Buenos Aires	Copa Claro	2013-506
2013	14	Marseille	Open 13	2013-496
2013	15	Memphis	U.S. National Indoor Tennis Championships	2013-402
2013	16	Acapulco	Abierto Mexicano	2013-807
2013	17	Delray Beach	International Championships	2013-499
2013	18	Dubai	Dubai Tennis Championships	2013-495
2013	19	Indian Wells	BNP Paribas Open	2013-404
2013	20	Miami	Sony Ericsson Open	2013-403
2013	21	Casablanca	Grand Prix Hassan II	2013-360
2013	22	Houston	U.S. Men's Clay Court Championships	2013-717
2013	23	Monte Carlo	Monte Carlo Masters	2013-410
2013	24	Barcelona	Open Banco Sabadell	2013-425
2013	25	Bucharest	BRD Nastase Tiriach Trophy	2013-773
2013	26	Munich	BMW Open	2013-308
2013	27	Oeiras	Portugal Open	2013-468
2013	28	Madrid	Mutua Madrid Open	2013-1536
2013	29	Rome	Internazionali BNL d'Italia	2013-416
2013	30	Dusseldorf	Power Horse Cup	2013-6710
2013	31	Nice	Open de Nice Côte d'Azur	2013-6120
2013	32	Paris	French Open	2013-520
2013	33	Halle	Gerry Weber Open	2013-500
2013	34	Queens Club	AEGON Championships	2013-311
2013	35	Eastbourne	AEGON International	2013-741
2013	36	's-Hertogenbosch	Topshelf Open	2013-440
2013	37	London	Wimbledon	2013-540
2013	38	Bastad	SkiStar Swedish Open	2013-316
2013	39	Newport	Hall of Fame Championships	2013-315
2013	40	Stuttgart	Mercedes Cup	2013-321
2013	41	Bogota	Claro Open Colombia	2013-6718
2013	42	Hamburg	German Open Tennis Championships	2013-414
2013	43	Atlanta	BB&T Atlanta Open	2013-6116
2013	44	Gstaad	Crédit Agricole Suisse Open Gstaad	2013-314
2013	45	Umag	ATP Vegeta Croatia Open	2013-439
2013	46	Kitzbuhel	Bet-At-Home Cup	2013-319
2013	47	Washington	Citi Open	2013-418
2013	48	Montreal	Rogers Masters	2013-421
2013	49	Cincinnati	Western & Southern Financial Group Masters	2013-422
2013	50	Winston-Salem	Winston-Salem Open at Wake Forest University	2013-6242
2013	51	New York	US Open	2013-560
2013	52	Metz	Open de Moselle	2013-341
2013	53	St. Petersburg	St. Petersburg Open	2013-568
2013	54	Bangkok	Thailand Open	2013-1720

2013	55	Kuala Lumpur	Malaysian Open	2013-6003
2013	56	Beijing	China Open	2013-747
2013	57	Tokyo	Rakuten Japan Open Tennis Championships	2013-329
2013	58	Shanghai	Shanghai Masters	2013-5014
2013	59	Moscow	Kremlin Cup	2013-438
2013	60	Stockholm	Stockholm Open	2013-429
2013	61	Vienna	Erste Bank Open	2013-337
2013	62	Basel	Swiss Indoors	2013-328
2013	63	Valencia	Valencia Open 500	2013-573
2013	64	Paris	BNP Paribas Masters	2013-352
2013	65	London	Masters Cup	2013-605
2014	1	Brisbane	Brisbane International	2014-339
2014	2	Chennai	Chennai Open	2014-891
2014	3	Doha	Qatar Exxon Mobil Open	2014-451
2014	4	Auckland	Heineken Open	2014-301
2014	5	Sydney	Apia International	2014-338
2014	6	Melbourne	Australian Open	2014-580
2014	7	Montpellier	Open Sud de France	2014-375
2014	8	Vina del Mar	Royal Guard Open Chile	2014-505
2014	9	Zagreb	PBZ Zagreb Indoors	2014-2276
2014	10	Buenos Aires	Copa Claro	2014-506
2014	11	Memphis	U.S. National Indoor Tennis Championships	2014-402
2014	12	Rotterdam	ABN AMRO World Tennis Tournament	2014-407
2014	13	Delray Beach	Delray Beach Open	2014-499
2014	14	Marseille	Open 13	2014-496
2014	15	Rio de Janeiro	Rio Open	2014-6932
2014	16	Acapulco	Abierto Mexicano	2014-807
2014	17	Dubai	Dubai Tennis Championships	2014-495
2014	18	Sao Paulo	Brasil Open	2014-533
2014	19	Indian Wells	BNP Paribas Open	2014-404
2014	20	Miami	Sony Ericsson Open	2014-403
2014	21	Casablanca	Grand Prix Hassan II	2014-360
2014	22	Houston	U.S. Men's Clay Court Championships	2014-717
2014	23	Monte Carlo	Monte Carlo Masters	2014-410
2014	24	Barcelona	Open Banco Sabadell	2014-425
2014	25	Bucharest	BRD Nastase Tiriatic Trophy	2014-773
2014	26	Munich	BMW Open	2014-308
2014	27	Oeiras	Portugal Open	2014-468
2014	28	Madrid	Mutua Madrid Open	2014-1536
2014	29	Rome	Internazionali BNL d'Italia	2014-416
2014	30	Dusseldorf	Dusseldorf Open	2014-6710
2014	31	Nice	Open de Nice Côte d'Azur	2014-6120
2014	32	Paris	French Open	2014-520
2014	33	Halle	Gerry Weber Open	2014-500
2014	34	Queens Club	AEGON Championships	2014-311



2014	35	Eastbourne	AEGON International	2014-741
2014	36	's-Hertogenbosch	Topshelf Open	2014-440
2014	37	London	Wimbledon	2014-540
2014	38	Bastad	SkiStar Swedish Open	2014-316
2014	39	Newport	Hall of Fame Championships	2014-315
2014	40	Stuttgart	Mercedes Cup	2014-321
2014	41	Bogota	Claro Open Colombia	2014-6718
2014	42	Hamburg	bet-at-home Open	2014-414
2014	43	Atlanta	BB&T Atlanta Open	2014-6116
2014	44	Gstaad	Crédit Agricole Suisse Open Gstaad	2014-314
2014	45	Umag	ATP Vegeta Croatia Open	2014-439
2014	46	Kitzbuhel	Bet-At-Home Cup	2014-319
2014	47	Washington	Citi Open	2014-418
2014	48	Toronto	Rogers Masters	2014-421
2014	49	Cincinnati	Western & Southern Financial Group Masters	2014-422
2014	50	Winston-Salem	Winston-Salem Open at Wake Forest University	2014-6242
2014	51	New York	US Open	2014-560
2014	52	Metz	Open de Moselle	2014-341
2014	53	Kuala Lumpur	Malaysian Open	2014-6003
2014	54	Shenzhen	Shenzhen Open	2014-6967
2014	55	Beijing	China Open	2014-747
2014	56	Tokyo	Rakuten Japan Open Tennis Championships	2014-329
2014	57	Shanghai	Shanghai Masters	2014-5014
2014	58	Moscow	Kremlin Cup	2014-438
2014	59	Stockholm	Stockholm Open	2014-429
2014	60	Vienna	Erste Bank Open	2014-337
2014	61	Basel	Swiss Indoors	2014-328
2014	62	Valencia	Valencia Open 500	2014-573
2014	63	Paris	BNP Paribas Masters	2014-352
2014	64	London	Masters Cup	2014-605
2015	1	Brisbane	Brisbane International	2015-339
2015	2	Chennai	Chennai Open	2015-891
2015	3	Doha	Qatar Exxon Mobil Open	2015-451
2015	4	Auckland	Heineken Open	2015-301
2015	5	Sydney	Apia International	2015-338
2015	6	Melbourne	Australian Open	2015-580
2015	7	Montpellier	Open Sud de France	2015-375
2015	8	Quito	Ecuador Open	2015-7161
2015	9	Zagreb	PBZ Zagreb Indoors	2015-2276
2015	10	Memphis	Memphis Open	2015-402
2015	11	Rotterdam	ABN AMRO World Tennis Tournament	2015-407
2015	12	Sao Paulo	Brasil Open	2015-533
2015	13	Delray Beach	Delray Beach Open	2015-499
2015	14	Marseille	Open 13	2015-496
2015	15	Rio de Janeiro	Rio Open	2015-6932

2015	16	Acapulco	Abierto Mexicano	2015-807
2015	17	Buenos Aires	Argentina Open	2015-506
2015	18	Dubai	Dubai Tennis Championships	2015-495
2015	19	Indian Wells	BNP Paribas Open	2015-404
2015	20	Miami	Sony Ericsson Open	2015-403
2015	21	Casablanca	Grand Prix Hassan II	2015-360
2015	22	Houston	U.S. Men's Clay Court Championships	2015-717
2015	23	Monte Carlo	Monte Carlo Masters	2015-410
2015	24	Barcelona	Open Banco Sabadell	2015-425
2015	25	Bucharest	BRD Nastase Tiriach Trophy	2015-773
2015	26	Estoril	Millenium Estoril Open	2015-7290
2015	27	Istanbul	Istanbul Open	2015-7163
2015	28	Munich	BMW Open	2015-308
2015	29	Madrid	Mutua Madrid Open	2015-1536
2015	30	Rome	Internazionali BNL d'Italia	2015-416
2015	31	Geneva	Geneva Open	2015-322
2015	32	Nice	Open de Nice Côte d'Azur	2015-6120
2015	33	Paris	French Open	2015-520
2015	34	's-Hertogenbosch	Topshelf Open	2015-440
2015	35	Stuttgart	Mercedes Cup	2015-321
2015	36	Halle	Gerry Weber Open	2015-500
2015	37	Queens Club	AEGON Championships	2015-311
2015	38	Nottingham	AEGON Open	2015-741
2015	39	London	Wimbledon	2015-540
2015	40	Newport	Hall of Fame Championships	2015-315
2015	41	Bastad	SkiStar Swedish Open	2015-316
2015	42	Bogota	Claro Open Colombia	2015-6718
2015	43	Umag	Konzum Croatia Open	2015-439
2015	44	Atlanta	BB&T Atlanta Open	2015-6116
2015	45	Gstaad	Suisse Open Gstaad	2015-314
2015	46	Hamburg	bet-at-home Open	2015-414
2015	47	Kitzbuhel	Generali Open	2015-319
2015	48	Washington	Citi Open	2015-418
2015	49	Montreal	Rogers Masters	2015-421
2015	50	Cincinnati	Western & Southern Financial Group Masters	2015-422
2015	51	Winston-Salem	Winston-Salem Open at Wake Forest University	2015-6242
2015	52	New York	US Open	2015-560
2015	53	Metz	Open de Moselle	2015-341
2015	54	St. Petersburg	St. Petersburg Open	2015-568
2015	55	Kuala Lumpur	Malaysian Open	2015-6003
2015	56	Shenzhen	Shenzhen Open	2015-6967
2015	57	Beijing	China Open	2015-747
2015	58	Tokyo	Rakuten Japan Open Tennis Championships	2015-329
2015	59	Shanghai	Shanghai Masters	2015-5014
2015	60	Moscow	Kremlin Cup	2015-438

2015	61	Stockholm	Stockholm Open	2015-429
2015	62	Vienna	Erste Bank Open	2015-337
2015	63	Basel	Swiss Indoors	2015-328
2015	64	Valencia	Valencia Open 500	2015-573
2015	65	Paris	BNP Paribas Masters	2015-352
2015	66	London	Masters Cup	2015-605
2016	1	Brisbane	Brisbane International	2016-M020
2016	2	Chennai	Chennai Open	2016-0891
2016	3	Doha	Qatar Exxon Mobil Open	2016-0451
2016	4	Auckland	ASB Classic	2016-0301
2016	5	Sydney	Apia International	2016-M001
2016	6	Melbourne	Australian Open	2016-580
2016	7	Montpellier	Open Sud de France	2016-0375
2016	8	Quito	Ecuador Open	2016-7161
2016	9	Sofia	Garanti Koza Sofia Open	2016-7434
2016	10	Buenos Aires	Argentina Open	2016-0506
2016	11	Memphis	Memphis Open	2016-0402
2016	12	Rotterdam	ABN AMRO World Tennis Tournament	2016-0407
2016	13	Delray Beach	Delray Beach Open	2016-0499
2016	14	Marseille	Open 13	2016-0496
2016	15	Rio de Janeiro	Rio Open	2016-M052
2016	16	Acapulco	Abierto Mexicano	2016-M004
2016	17	Dubai	Dubai Tennis Championships	2016-0495
2016	18	Sao Paulo	Brasil Open	2016-0533
2016	19	Indian Wells	BNP Paribas Open	2016-M006
2016	20	Miami	Sony Ericsson Open	2016-M007
2016	21	Houston	U.S. Men's Clay Court Championships	2016-0717
2016	22	Marrakech	Grand Prix Hassan II	2016-0360
2016	23	Monte Carlo	Monte Carlo Masters	2016-0410
2016	24	Barcelona	Open Banco Sabadell	2016-0425
2016	25	Bucharest	BRD Nastase Tiriach Trophy	2016-0773
2016	26	Estoril	Millenium Estoril Open	2016-7290
2016	27	Istanbul	Istanbul Open	2016-7163
2016	28	Munich	BMW Open	2016-0308
2016	29	Madrid	Mutua Madrid Open	2016-M021
2016	30	Rome	Internazionali BNL d'Italia	2016-M009
2016	31	Geneva	Geneva Open	2016-0322
2016	32	Nice	Open de Nice Côte d'Azur	2016-6120
2016	33	Paris	French Open	2016-520
2016	34	's-Hertogenbosch	Topshelf Open	2016-M010
2016	35	Stuttgart	Mercedes Cup	2016-0321
2016	36	Halle	Gerry Weber Open	2016-0500
2016	37	Queens Club	AEGON Championships	2016-0311
2016	38	Nottingham	AEGON Open	2016-0741
2016	39	London	Wimbledon	2016-540

2016	40	Bastad	SkiStar Swedish Open	2016-0316
2016	41	Hamburg	German Tennis Championships	2016-0414
2016	42	Newport	Hall of Fame Championships	2016-0315
2016	43	Gstaad	Suisse Open Gstaad	2016-0314
2016	44	Kitzbuhel	Generali Open	2016-0319
2016	45	Umag	Konzum Croatia Open	2016-0439
2016	46	Washington	Citi Open	2016-M035
2016	47	Toronto	Rogers Masters	2016-0421
2016	48	Atlanta	BB&T Atlanta Open	2016-6116
2016	49	Los Cabos	Abierto Mexicano Mifel	2016-7480
2016	50	Cincinnati	Western & Southern Financial Group Masters	2016-M024
2016	51	Winston-Salem	Winston-Salem Open at Wake Forest University	2016-6242
2016	52	New York	US Open	2016-560
2016	53	Metz	Open de Moselle	2016-0341
2016	54	St. Petersburg	St. Petersburg Open	2016-0568
2016	55	Chengdu	Chengdu Open	2016-7581
2016	56	Shenzhen	Shenzhen Open	2016-6967
2016	57	Beijing	China Open	2016-M015
2016	58	Tokyo	Rakuten Japan Open Tennis Championships	2016-0329
2016	59	Shanghai	Shanghai Masters	2016-5014
2016	60	Antwerp	European Open	2016-7485
2016	61	Moscow	Kremlin Cup	2016-M014
2016	62	Stockholm	Stockholm Open	2016-0429
2016	63	Basel	Swiss Indoors	2016-0328
2016	64	Vienna	Erste Bank Open	2016-0337
2016	65	Paris	BNP Paribas Masters	2016-0352
2016	66	London	Masters Cup	2016-0605
2017	1	Brisbane	Brisbane International	2017-M020
2017	2	Chennai	Chennai Open	2017-0891
2017	3	Doha	Qatar Exxon Mobil Open	2017-0451
2017	4	Auckland	ASB Classic	2017-0301
2017	5	Sydney	Apia International	2017-M001
2017	6	Melbourne	Australian Open	2017-580
2017	7	Montpellier	Open Sud de France	2017-0375
2017	8	Quito	Ecuador Open	2017-7161
2017	9	Sofia	Garanti Koza Sofia Open	2017-7434
2017	10	Buenos Aires	Argentina Open	2017-0506
2017	11	Memphis	Memphis Open	2017-0402
2017	12	Rotterdam	ABN AMRO World Tennis Tournament	2017-0407
2017	13	Delray Beach	Delray Beach Open	2017-0499
2017	14	Marseille	Open 13	2017-0496
2017	15	Rio de Janeiro	Rio Open	2017-6932
2017	16	Acapulco	Abierto Mexicano	2017-M004
2017	17	Dubai	Dubai Tennis Championships	2017-0495
2017	18	Sao Paulo	Brasil Open	2017-0533

2017	19	Indian Wells	BNP Paribas Open	2017-M006
2017	20	Miami	Sony Ericsson Open	2017-M007
2017	21	Houston	U.S. Men's Clay Court Championships	2017-0717
2017	22	Marrakech	Grand Prix Hassan II	2017-0360
2017	23	Monte Carlo	Monte Carlo Masters	2017-0410
2017	24	Barcelona	Open Banco Sabadell	2017-0425
2017	25	Budapest	Gazprom Hungarian Open	2017-7648
2017	26	Estoril	Millenium Estoril Open	2017-7290
2017	27	Istanbul	Istanbul Open	2017-7163
2017	28	Munich	BMW Open	2017-0308
2017	29	Madrid	Mutua Madrid Open	2017-M021
2017	30	Rome	Internazionali BNL d'Italia	2017-M009
2017	31	Geneva	Geneva Open	2017-0322
2017	32	Lyon	Lyon Open	2017-7694
2017	33	Paris	French Open	2017-520
2017	34	's-Hertogenbosch	Ricoh Open	2017-M010
2017	35	Stuttgart	Mercedes Cup	2017-0321
2017	36	Halle	Gerry Weber Open	2017-0500
2017	37	Queens Club	AEGON Championships	2017-0311
2017	38	Antalya	Antalya Open	2017-7650
2017	39	Eastbourne	AEGON International	2017-M016
2017	40	London	Wimbledon	2017-540
2017	41	Bastad	SkiStar Swedish Open	2017-0316
2017	42	Newport	Hall of Fame Championships	2017-0315
2017	43	Umag	Croatia Open	2017-0439
2017	44	Atlanta	BB&T Atlanta Open	2017-6116
2017	45	Gstaad	Suisse Open Gstaad	2017-0314
2017	46	Hamburg	German Tennis Championships	2017-0414
2017	47	Kitzbuhel	Generali Open	2017-0319
2017	48	Los Cabos	Abierto Mexicano Mifel	2017-7480
2017	49	Washington	Citi Open	2017-M035
2017	50	Montreal	Rogers Masters	2017-0421
2017	51	Cincinnati	Western & Southern Financial Group Masters	2017-M024
2017	52	Winston-Salem	Winston-Salem Open at Wake Forest University	2017-6242
2017	53	New York	US Open	2017-560
2017	54	Metz	Open de Moselle	2017-0341
2017	55	St. Petersburg	St. Petersburg Open	2017-0568
2017	56	Chengdu	Chengdu Open	2017-7581
2017	57	Shenzhen	Shenzhen Open	2017-6967
2017	58	Beijing	China Open	2017-M015
2017	59	Tokyo	Rakuten Japan Open Tennis Championships	2017-0329
2017	60	Shanghai	Shanghai Masters	2017-5014
2017	61	Antwerp	European Open	2017-7485
2017	62	Moscow	Kremlin Cup	2017-M014
2017	63	Stockholm	Stockholm Open	2017-0429

2017	64	Basel	Swiss Indoors	2017-0328
2017	65	Vienna	Erste Bank Open	2017-0337
2017	66	Paris	BNP Paribas Masters	2017-0352
2017	67	London	Masters Cup	2017-0605
2018	1	Brisbane	Brisbane International	2018-M020
2018	2	Doha	Qatar Exxon Mobil Open	2018-0451
2018	3	Pune	Tata Open	2018-0891
2018	4	Auckland	ASB Classic	2018-0301
2018	5	Sydney	Sydney International	2018-M001
2018	6	Melbourne	Australian Open	2018-580
2018	7	Montpellier	Open Sud de France	2018-0375
2018	8	Quito	Ecuador Open	2018-7161
2018	9	Sofia	Garanti Koza Sofia Open	2018-7434
2018	10	Buenos Aires	Argentina Open	2018-0506
2018	11	New York	New York Open	2018-0424
2018	12	Rotterdam	ABN AMRO World Tennis Tournament	2018-0407
2018	13	Delray Beach	Delray Beach Open	2018-0499
2018	14	Marseille	Open 13	2018-0496
2018	15	Rio de Janeiro	Rio Open	2018-6932
2018	16	Acapulco	Abierto Mexicano	2018-M004
2018	17	Dubai	Dubai Tennis Championships	2018-0495
2018	18	Sao Paulo	Brasil Open	2018-0533
2018	19	Indian Wells	BNP Paribas Open	2018-M006
2018	20	Miami	Sony Ericsson Open	2018-M007
2018	21	Houston	U.S. Men's Clay Court Championships	2018-0717
2018	22	Marrakech	Grand Prix Hassan II	2018-0360
2018	23	Monte Carlo	Monte Carlo Masters	2018-0410
2018	24	Barcelona	Open Banco Sabadell	2018-0425
2018	25	Budapest	Gazprom Hungarian Open	2018-7648
2018	26	Estoril	Millennium Estoril Open	2018-7290
2018	27	Istanbul	Istanbul Open	2018-7163
2018	28	Munich	BMW Open	2018-0308
2018	29	Madrid	Mutua Madrid Open	2018-M021
2018	30	Rome	Internazionali BNL d'Italia	2018-M009
2018	31	Geneva	Geneva Open	2018-0322
2018	32	Lyon	Lyon Open	2018-7694
2018	33	Paris	French Open	2018-520
2018	34	's-Hertogenbosch	Ricoh Open	2018-M010
2018	35	Stuttgart	Mercedes Cup	2018-0321
2018	36	Halle	Gerry Weber Open	2018-0500
2018	37	Queens Club	AEGON Championships	2018-0311
2018	38	Antalya	Antalya Open	2018-7650
2018	39	Eastbourne	Eastbourne International	2018-M016
2018	40	London	Wimbledon	2018-540
2018	41	Bastad	SkiStar Swedish Open	2018-0316

2018	42	Newport	Hall of Fame Championships	2018-0315
2018	43	Umag	Croatia Open	2018-0439
2018	44	Atlanta	BB&T Atlanta Open	2018-6116
2018	45	Gstaad	Suisse Open Gstaad	2018-0314
2018	46	Hamburg	German Tennis Championships	2018-0414
2018	47	Kitzbuhel	Generali Open	2018-0319
2018	48	Los Cabos	Abierto Mexicano Mifel	2018-7480
2018	49	Washington	Citi Open	2018-M035
2018	50	Toronto	Rogers Masters	2018-0421
2018	51	Cincinnati	Western & Southern Financial Group Masters	2018-M024
2018	52	Winston-Salem	Winston-Salem Open at Wake Forest University	2018-6242
2018	53	New York	US Open	2018-560
2018	54	Metz	Open de Moselle	2018-0341
2018	55	St. Petersburg	St. Petersburg Open	2018-0568
2018	56	Chengdu	Chengdu Open	2018-7581
2018	57	Shenzhen	Shenzhen Open	2018-6967
2018	58	Beijing	China Open	2018-M015
2018	59	Tokyo	Rakuten Japan Open Tennis Championships	2018-0329
2018	60	Shanghai	Shanghai Masters	2018-5014
2018	61	Antwerp	European Open	2018-7485
2018	62	Moscow	Kremlin Cup	2018-M014
2018	63	Stockholm	Stockholm Open	2018-0429
2018	64	Basel	Swiss Indoors	2018-0328
2018	65	Vienna	Erste Bank Open	2018-0337
2018	66	Paris	BNP Paribas Masters	2018-0352
2018	67	London	Masters Cup	2018-0605
2019	1	Brisbane	Brisbane International	2019-M020
2019	2	Doha	Qatar Exxon Mobil Open	2019-0451
2019	3	Pune	Maharashtra Open	2019-0891
2019	4	Auckland	ASB Classic	2019-0301
2019	5	Sydney	Sydney International	2019-M001
2019	6	Melbourne	Australian Open	2019-580
2019	7	Cordoba	Cordoba Open	2019-9158
2019	8	Montpellier	Open Sud de France	2019-0375
2019	9	Sofia	Sofia Open	2019-7434
2019	10	Buenos Aires	Argentina Open	2019-0506
2019	11	New York	New York Open	2019-0424
2019	12	Rotterdam	ABN AMRO World Tennis Tournament	2019-0407
2019	13	Delray Beach	Delray Beach Open	2019-0499
2019	14	Marseille	Open 13	2019-0496
2019	15	Rio de Janeiro	Rio Open	2019-6932
2019	16	Acapulco	Abierto Mexicano	2019-M004
2019	17	Dubai	Dubai Tennis Championships	2019-0495
2019	18	Sao Paulo	Brasil Open	2019-0533
2019	19	Indian Wells	BNP Paribas Open	2019-M006

2019	20	Miami	Sony Ericsson Open	2019-M007
2019	21	Houston	U.S. Men's Clay Court Championships	2019-0717
2019	22	Marrakech	Grand Prix Hassan II	2019-0360
2019	23	Monte Carlo	Monte Carlo Masters	2019-0410
2019	24	Barcelona	Open Banco Sabadell	2019-0425
2019	25	Budapest	Hungarian Open	2019-7648
2019	26	Estoril	Millennium Estoril Open	2019-7290
2019	27	Munich	BMW Open	2019-0308
2019	28	Madrid	Mutua Madrid Open	2019-M021
2019	29	Rome	Internazionali BNL d'Italia	2019-M009
2019	30	Geneva	Geneva Open	2019-0322
2019	31	Lyon	Lyon Open	2019-7694
2019	32	Paris	French Open	2019-520
2019	33	's-Hertogenbosch	Rosmalen Grass Court Championships	2019-M010
2019	34	Stuttgart	Mercedes Cup	2019-0321
2019	35	Halle	Halle Open	2019-0500
2019	36	Queens Club	Queen's Club Championships	2019-0311
2019	37	Antalya	Antalya Open	2019-7650
2019	38	Eastbourne	Eastbourne International	2019-M016
2019	39	London	Wimbledon	2019-540
2019	40	Bastad	SkiStar Swedish Open	2019-0316
2019	41	Newport	Hall of Fame Championships	2019-0315
2019	42	Umag	Croatia Open	2019-0439
2019	43	Atlanta	BB&T Atlanta Open	2019-6116
2019	44	Gstaad	Suisse Open Gstaad	2019-0314
2019	45	Hamburg	German Tennis Championships	2019-0414
2019	46	Kitzbuhel	Generali Open	2019-0319
2019	47	Los Cabos	Abierto Mexicano Mifel	2019-7480
2019	48	Washington	Citi Open	2019-M035
2019	49	Montreal	Rogers Masters	2019-0421
2019	50	Cincinnati	Western & Southern Financial Group Masters	2019-M024
2019	51	Winston-Salem	Winston-Salem Open at Wake Forest University	2019-6242
2019	52	New York	US Open	2019-560
2019	53	Metz	Open de Moselle	2019-0341
2019	54	St. Petersburg	St. Petersburg Open	2019-0568
2019	55	Chengdu	Chengdu Open	2019-7581
2019	56	Zhuhai	Zhuhai Open	2019-9164
2019	57	Beijing	China Open	2019-M015
2019	58	Tokyo	Rakuten Japan Open Tennis Championships	2019-0329
2019	59	Shanghai	Shanghai Masters	2019-5014
2019	60	Antwerp	European Open	2019-7485
2019	61	Moscow	Kremlin Cup	2019-M014
2019	62	Stockholm	Stockholm Open	2019-0429
2019	63	Basel	Swiss Indoors	2019-0328
2019	64	Vienna	Erste Bank Open	2019-0337



2019	65	Paris	BNP Paribas Masters	2019-0352
2019	66	London	Masters Cup	2019-0605

### DE-PARA de Jogadores

Al Mutawa J.	106325
Al-Alawi S.K.	102956
Al-Ghareeb M.	103600
Alawadhi O.	200525
Albert M.	103988
Ali Mutawa J.M.	106325
Alvarez E.	102135
Ancic I.	104339
Andersen J.F.	102107
Aragone JC.	111574
Aranguren J.M.	104253
Artunedo Martinavarro A.	106238
Ascione A.	103693
Auger-Aliassime F.	200000
Bachelot J.F.	103003
Bahrouzyan O.	103914
Bautista Agut R.	105138
Bautista R.	105138
Benneteau A.	103898
Bogomolov A.	104166
Bogomolov Jr. A.	104166
Bogomolov Jr.A.	104166
Brugues-Davi A.	104516
Brzezicki J.P.	103976
Cabal J.S.	108701
Carballes Baena R.	106148
Carreno Busta P.	105807
Carreno-Busta P.	105807
Cervantes I.	105438
Chekov P.	105189
Chela J.	103428
Chela J.I.	103428
Dasnieres de Veigy J.	104850
Davidovich Fokina A.	200221
De Bakker T.	105217
De Chaunac S.	103063
De Greef A.	105963
De Heart R.	104281
De Loore J.	106167

De Minaur A.	200282
De Paula F.	105245
De Schepper K.	104932
De Voest R.	103573
Deen Heshaam A.	106277
Del Bonis F.	105643
Del Potro J.	105223
Del Potro J. M.	105223
Del Potro J.M.	105223
Dell'Acqua M.	103435
Dev Varman S.	104500
Di Mauro A.	103035
Di Pasquale A.	103324
Do M.Q.	104283
Dolgoplov O.	105238
Duclos P.L.	104661
Dutra Da Silva R.	104297
Dutra Silva R.	104297
El Aarej M.	103007
El Amrani R.	105149
El Aynaoui Y.	101962
Estrella Burgos V.	103607
Faurel J.C.	103726
Ferrero J.	103507
Ferrero J.C.	103507
Fish A.	103888
Fornell M.	103924
Fruttero J.P.	103762
Fruttero J. P.	103762
Galan D.E.	123755
Gallardo M.	103746
Gallardo Valles M.	103746
Gambill J.M.	102998
Garcia-Lopez G.	104198
Garcia-Sintes J.	103365
Gimeno D.	104593
Gimeno-Traver D.	104593
Goellner M.K.	101806
Gomez-Herrera C.	105529
Gong M.X.	105001
Granollers Pujol G.	105283
Granollers-Pujol G.	105283
Granollers-Pujol M.	105283
Gromley C.	-1
Gruber K.	103460

Guccione A.	104589
Gutierrez-Ferrol S.	105300
Guzman J.	103700
Guzman J.P.	103700
Haider-Mauer A.	104890
Haider-Maurer A.	104890
Haji A.	108993
Hantschek M.	103082
Harper-Griffith L.	103834
Herbert P-H.	105732
Herbert P.H.	105732
Herm-Zahlava J.	103564
Hernandez-Fernandez J.	105497
Huesler M.A.	144817
Huta Galung J.	104619
Im K.T.	103682
Ivanov-Smolensky K.	103715
Jeong S.Y.	106158
Jones G.D.	110685
Jun W.	104753
Jun W.S.	104753
King-Turner D.	104368
Kohlschreiber P..	104259
Kucera V.	102344
Kunitcin I.	103857
Kutac R.	104519
Kuznetsov Al.	104864
Kuznetsov An.	105723
Kwiatkowski T.S.	111515
Kwon S.W.	126952
Lee D.H.	117353
Lee H.T.	102703
Lin J.M.	200368
Lisnard J.	103444
Lisnard J.R.	103444
Londero J.I.	106228
Lopez Villasenor G.	126204
Lopez-Jaen M.A.	104091
Lopez-Perez E.	105782
Lu Y.	104229
Lu Y.H.	104229
Luncanu P.A.	105331
Luque D.	104317
March O.	201995
Marin J.A.	102548

Marin L.	102192
Mathieu P.	103908
Mathieu P.H.	103908
Matos-Gil I.	104099
Mecir M.	105080
Menendez-Maceiras A.	104629
Monteiro J.	106329
Moroni G.	126237
Munoz De La Nava D.	103926
Munoz de La Nava D.	103926
Munoz de la Nava D.	103926
Munoz-De La Nava D.	103926
Nadal-Parera R.	104745
Nader M.	108996
Nam H.W.	104478
Navarro-Pastor I.	103868
Nedovyesov O.	104873
O'Brien A.	101711
O'Neal J.	103076
Ojeda Lara R.	106120
Ortega-Olmedo R.	105751
Podlipnik H.	105070
Prashanth V.	104822
Prpic A.	104004
Querry S.	105023
Qureshi A.	103529
Qureshi A.U.H.	103529
Ramirez Hidalgo R.	103105
Ramirez-Hidalgo R.	103105
Ramos-Vinolas A.	105077
Rascon T.	101869
Reyes-Varela M.A.	104960
Riba-Madrid P.	105137
Robredo R.	103990
Roca Batalla O.	106177
Roger-Vasselin E.	104273
Ruevski P.	104234
Saavedra Corvalan C.	105601
Salva B.	104831
Samper-Montana J.	105515
Sanchez De Luna J.	104445
Sanchez de Luna J.A.	104445
Scherrer J.	103238
Scherrer J.C.	103238
Schuettler P.	102783

Schuttler P.	102783
Seyboth Wild T.	205734
Si Y.M.	103427
Silva D.	105180
Silva F.	106393
Silva F.F.	106393
Smith J.P.	105441
Statham J.	104907
Statham R.	104907
Stebe C-M.	105649
Stebe C.M.	105649
Stepanek M.	103285
Struff J-L.	105526
Struff J.L.	105526
Sultan-Khalfan A.	102956
Trujillo G.	103448
Tseng C.H.	202358
Tsonga J.W.	104542
Tyurnev E.	126185
Van D. Merwe I.	104292
Van Der Merwe I.	104292
Van Lottum J.	102770
Van Rijthoven T.	126646
Van der Dium A.	105014
Van der Merwe I.	104292
Vassallo-Arguello M.	103506
Vega Hernandez D.	110754
Verdasco M.	104269
Vicente M.	102950
Viloca J.A.	102167
Viola Mat.	104970
Wang Y.	103817
Wang Y. Jr.	103817
Wang Y.Jr.	103817
Wang Y.T.	103817
Xu J.C.	105220
Yang T.H.	105738
Yoon Y.	102263
Youzhny A.	104022
Yu X.Y.	104501
Zapata Miralles B.	126523
Zayed M.S.	110476
Zayid M.S.	122570
Zeng S.X.	103830
Zhang Ze.	105585

Zhang Zh.	105585
Zhu B.Q.	103337
de Chaunac S.	103063
de Voest R.	103573
di Mauro A.	103035
di Pasquale A.	103324
van Gemerden M.	103369
van Lottum J.	102770
van Scheppingen D.	102615
van der Meer N.	104485
Svajda Z.	208260
Kachmazov A.	208230
Nava E.	207182
Jubb P.	206703
Sinner J.	206173
Gray A.	202400
Brooksby J.	202385
Moeller M.	202359
Reinberg E.	202130
Cerundolo F.	202103
Andreev A.	202090
Korda S.	200624
Popyrin A.	200615
Benchetrit E.	200611
Donski A.	200574
Avidzba A.	200553
Kypson P.	200516
Rodionov J.	200514
Molleker R.	200484
Piros Z.	200436
Gaston H.	200384
Lazarov A.	200247
Valkusz M.	200218
Kecmanovic M.	200175
Kuhn N.	200095
Wu Y.	200059
Catarina L.	200031
Humbert U.	200005
Kirkin E.	144985
Dubrivnyy A.	144984
Blancaneaux G.	144972
Roumane R.	144946
Wessels L.	144923
Moutet C.	144895
Soderlund C.	144797

Ahouda A.	144760
Harris L.	144750
Munar J.	144719
Ymer M.	144707
Kolar Z.	144645
Koepfer D.	136440
Griekspoor T.	134868
Ruud C.	134770
Bellier A.	134472
Shapovalov D.	133430
Watanuki Y.	133297
Caruana L.	132482
Sonego L.	132283
Hurkacz H.	128034
Gojo B.	127339
Altmaier D.	127157
Vukic A.	126846
Purcell M.	126845
Tsitsipas S.	126774
Clarke J.	126652
Nikles J.	126627
Berrettini M.	126610
Taberner C.	126535
Takahashi Y.	126208
Tiafoe F.	126207
Paul T.	126205
Fritz T.	126203
Luz O.	126197
Hoang A.	126156
Shane R.	126152
Mager G.	126149
Safiullin R.	126128
Bonzi B.	126127
Janvier M.	126125
Serdarusic N.	126120
Rublev A.	126094
Ivashka I.	125802
Opelka R.	124187
Muller A.	124186
Ofner S.	124116
Martinez P.	124079
Escobedo E.	124014
Hemery C.	123921
Choinski J.	123828
Celikbilek A.	123795

Altamirano C.	123768
Zayid M. S.	122570
Leshem E.	122548
Bublik A.	122330
Bogaerts R.	122279
Moraing M.	121531
Maden Y.	120424
Chrysochos P.	117365
Santillan A.	117361
Polmans M.	117360
Jasika O.	117357
Ilkel C.	117356
Anderson O.	117352
Norrie C.	111815
Jarry N.	111797
Baldi F.	111795
Majchrzak K.	111794
Schnur B.	111790
Mmoh M.	111581
Kozlov S.	111578
Donaldson J.	111577
Nagal S.	111576
Khachanov K.	111575
Aragone J.	111574
Djere L.	111513
Rubin N.	111511
Hamou M.	111506
Halys Q.	111460
Kadhe A.	111459
Mcdonald M.	111456
Torpegaard M.	111453
Hossam K.	111444
Thompson J.	111442
Chung H.	106184
Ymer E.	111200
Miedler L.	111192
Zhang Z.	105585
Pasha N.	111156
Eubanks C.	111153
Milojevic N.	110748
Tearney F.	110686
Thompson C.	110636
Novak D.	110602
Simonsson F.	110596
King K.	110536



Zayed M. S.	110476
Marterer M.	109739
Lenz J.	109699
Basso A.	109640
Napolitano S.	109303
Masur D.	109054
Bowles C.	108997
Hajji A.	108993
Gomez A.	108716
Koderisch C.	106764
Coric B.	106432
Ciorcila P.	106430
Garin C.	106426
Mott B.	106425
Kokkinakis T.	106423
Medvedev D.	106421
Quinzi G.	106420
Nishioka Y.	106415
Gomez L.	106412
Kyrgios N.	106401
Cachin P.	106398
Trongcharoenchaikul W.	106397
Donati M.	106390
Edmund K.	106378
Veger F.	106374
Ramanathan R.	106368
Lokoli L.	106362
Pavlasek A.	106361
Krstin P.	106353
Karlovskiy E.	106348
Halebian A.	106335
O'Connell C.	106331
Monteiro T.	106329
Harrison C.	106328
Safranek V.	106324
Gao X.	106321
Tepavac M.	106311
Pouille L.	106298
Barrere G.	106296
Coppejans K.	106293
Saville L.	106291
Peliwo F.	106290
Bourgue M.	106288
Krueger M.	106283
Broady L.	106281

Smith A.	106269
Novikov D.	106261
Patino L.	106250
Domingues J.	106249
Kern R.	106246
Golding O.	106244
Delic M.	106243
Karatsev A.	106234
Thiem D.	106233
Whittington A.	106225
Kuzmanov D.	106220
Giron M.	106218
Fratangelo B.	106216
Otte O.	106214
Vesely J.	106210
Pavic M.	106207
Sakamoto P.	106203
Dellien H.	106198
Kubler J.	106186
Lama G.	106174
Shyla Y.	106138
Lamasine T.	106137
Daniel T.	106121
Horansky F.	106110
Bolt A.	106109
Clezar G.	106105
Caruso S.	106099
Mitchell B.	106093
Berta D.	106090
Gerasimov E.	106078
Kovalik J.	106075
Vatutin A.	106072
Tomic B.	106071
Baluda V.	106066
Cecchinato M.	106065
Sock J.	106058
Davydenko P.	106053
Kudla D.	106045
Kicker N.	106044
Schwartzman D.	106043
Arvidsson I.	106035
Uchiyama Y.	106034
Arguello F.	106032
Sarmiento R.	106031
Bhambri Y.	106026

Velotti A.	106006
Lestienne C.	106005
Dzumhur D.	106000
Ouyang B.	105999
Harrison R.	105992
Biryukov M.	105987
King D.	105985
Laaksonen H.	105967
King E.	105960
Olivo R.	105952
Coria F.	105948
Gaio F.	105943
Rios F.	105941
Krajinovic F.	105936
Wang C.	105934
Quiroz R.	105933
Basilashvili N.	105932
Kosakowski D.	105930
Mina G.	105917
Fucsovics M.	105916
Krawietz K.	105906
Duckworth J.	105902
Marti J.	105894
Androic T.	105888
Travaglia S.	105882
Barton M.	105881
Gomez E.	105877
Olivetti A.	105874
Lindell C.	105872
Bangoura S.	105871
Hanfmann Y.	105870
Hank T.	105853
Wu D.	105842
Giustino L.	105841
Djokovic M.	105831
Khaddari H.	105830
Grigelis L.	105827
Bellotti R.	105821
Andreozzi G.	105819
Sandgren T.	105815
Basic M.	105806
Buchanan C.	105783
Vasilevski A.	105781
Dimitrov G.	105777
Ehrat S.	105757

Williams R.	105733
Diez S.	105731
Britton D.	105730
Kuznetsov A.	104864
Vovk S.	105720
Marcan D.	105711
Scholtz N.	105707
Bega A.	105692
Sanchez M.	105688
Raonic M.	105683
Struvay E.	105679
Michon A.	105678
Cacic N.	105677
Goffin D.	105676
Elias G.	105671
Janowicz J.	105668
Copil M.	105657
Arevalo M.	105656
Moriya H.	105655
Smethurst D.	105651
Willis M.	105650
Delbonis F.	105643
Rola B.	105641
Cox D.	105638
Galovic V.	105634
Safwat M.	105633
Garanganga T.	105626
Klahn B.	105614
Gombos N.	105613
Belyaev I.	105609
Satral J.	105596
Reid M.	105591
Ignatik U.	105590
Melzer G.	105589
Lazov A.	105586
Lajovic D.	105583
Wachiramanowong K.	105579
Pospisil V.	105577
Berankis R.	105575
Kontinen H.	105573
Krajicek A.	105572
Giannessi A.	105561
Eysseric J.	105559
Rufin G.	105558
Evans D.	105554

Yuksel A.	105551
Pella G.	105550
Trinker B.	105542
Donskoy E.	105539
Ward A.	105528
Sarkissian A.	105512
Wagh A.	105501
Bagnis F.	105487
Trungelliti M.	105477
Langer N.	105472
Ramirez C.	105471
Klein B.	105454
Nishikori K.	105453
Johnson S.	105449
Zlatanovic A.	105440
Eriksson M.	105436
Gunneswaran P.	105432
Albot R.	105430
Sekulic M.	105427
Linzer M.	105426
Martin A.	103252
Marcora R.	105406
McClune M.	105404
Mcclune M.	105404
Siriluethaiwattana P.	105402
Trevisan M.	105401
Young D.	105385
Bedene A.	105379
Gojowczyk P.	105376
Klizan M.	105373
Jung J.	105359
Millman J.	105357
Fabbiano T.	105341
Bai Y.	105339
Meister N.	105336
Paire B.	105332
Gigounon G.	105325
Schoorel T.	105318
Sousa J.	105311
Pavic A.	105302
Lajola D.	105294
Gonzalez A.	105292
Jones G.	104711
Granollers G.	105283
Quigley E.	105281

Escobar G.	105271
Teixeira M.	105270
Demoliner M.	105269
Romboli F.	105262
Mektic N.	105254
Ittogi T.	105252
Dolgoplov A.	105238
Nedunchezhiyan J.	105233
Belic L.	105231
Bester P.	105229
Cilic M.	105227
Balazs A.	105226
Sugita Y.	105216
Gulbis E.	105208
Lindahl N.	105194
Zekic M.	105181
Mesaros K.	105175
Mannarino A.	105173
Andersen J.	105169
Polansky P.	105166
Sakharov G.	105163
Marti Y.	105161
Sousa P.	105155
Souza J.	105154
Ito T.	105147
Riba P.	105137
Zopp J.	105132
Goransson T.	105131
Olaso G.	105128
Lejnieks K.	105122
Molteni A.	105120
Lojda D.	105116
Rosenholm P.	105113
Roshardt R.	105104
Corrie E.	105100
Veic A.	105099
Sidorenko A.	105098
Korolev E.	105095
Torebko P.	105091
Lemke J.	105084
Sergeyev I.	105079
Ramos A.	105077
Bemelmans R.	105074
Singh S.	105072
Panfil G.	105066

Smyczek T.	105065
Bellucci T.	105064
Arnaboldi A.	105063
Kukushkin M.	105062
Setkic A.	105060
Takeuchi K.	105059
Giraldo S.	105053
Eaton C.	105052
Ebden M.	105051
Clayton A.	105048
Puetz T.	105047
Lacko L.	105041
Groth S.	105032
Myneni S.	105031
Venus M.	105030
Betov S.	105029
Levine J.	105028
Niki T.	105026
Querrey S.	105023
Skugor F.	105015
Marchenko I.	105011
Kutrovsky D.	105004
Zverev M.	104999
Sijsling I.	104997
Vaisse M.	104993
Vardhan V.	104986
Golubev A.	104979
Brands D.	104978
Sweeting R.	104976
Kirillov E.	104975
Strode B.	104971
Viola M.	104970
Sadecky A.	104967
Cervenak P.	104965
Mertens Y.	104963
Ball C.	104958
Fidirko N.	104953
Thron A.	104951
Jelenic P.	104948
Ilhan M.	104947
Mcgee J.	104945
Desein N.	104944
Fognini F.	104926
Djokovic N.	104925
Mayer L.	104919

Murray A.	104918
Eleskovic E.	104914
Barrientos N.	104905
Haase R.	104898
Bachinger M.	104897
Agostinelli B.	104896
Authom M.	104893
Kavcic B.	104882
Nedovyesov A.	104873
Chardy J.	104871
Bubka S.	104869
Ward J.	104868
Farah R.	104857
Naso G.	104851
Bossel A.	104847
Idmbarek Y.	104839
Marrai M.	104830
Rodriguez B.	104823
Chvojka E.	104820
Rastogi K.	104816
Zemlja G.	104813
Jenkins S.	104812
Li Z.	104810
Ziadi M.	104807
Weintraub A.	104804
Rachidi Y.	104803
Martin F.	104802
Istomin D.	104797
Monfils G.	104792
Sabry S.	104789
Ledovskikh M.	104783
Ito Y.	104781
Baker J.	104779
Kumantsov A.	104778
Fischer M.	104770
Oudsema S.	104760
Gasquet R.	104755
Nadal R.	104745
Alcaide G.	104739
Zampieri C.	104738
Dustov F.	104736
Kamke T.	104735
Simmonds P.	104732
Anderson K.	104731
Nedelko I.	104729



Schwank E.	104724
Ouanna J.	104721
Granollers M.	104719
Evans B.	104717
Reister J.	104714
Matsukevitch D.	104707
Wolmarans F.	104695
Inglot D.	104693
Slabinsky A.	104692
Singh N.	104684
Rieschick S.	104681
Troicki V.	104678
Sitak A.	104677
Beck A.	104676
Millot V.	104672
Oswald P.	104667
Andujar P.	104665
Stakhovsky S.	104660
Cuevas P.	104655
Alund M.	104651
Raja P.	104646
Ghedin R.	104644
Galdon P.	104640
Odesnik W.	104639
Smeets R.	104635
Rodrigues M.	104633
Kudryavtsev A.	104628
Westerhof B.	104627
Goodall J.	104623
Bolelli S.	104620
Yoo D.	104612
Berdych T.	104607
Almagro N.	104597
Kloer S.	104595
Matosevic M.	104594
Bozoljac I.	104592
Guccione C.	104589
Rosol L.	104586
Jouan R.	104579
Baghdatis M.	104571
Vanni L.	104563
Gabashvili T.	104559
Abdulla M.	104558
Hocevar R.	104550
Baker B.	104548

Zeballos H.	104547
Isner J.	104545
Garza D.	104535
Sela D.	104534
Wawrinka S.	104527
Gil F.	104523
Montcourt M.	104515
Yim R.	104508
Kravchuk K.	104505
Hutchins R.	104504
Hormazabal G.	104503
Devvarman S.	104500
Wang J.	104499
Mergea F.	104495
Ungur A.	104494
Sandbichler A.	104488
Tecau H.	104486
Aguilar J.	104477
Sorensen L.	104476
Trusendi W.	104473
Dodig I.	104471
Tveit E.	104469
Simon G.	104468
Ouahab L.	104467
Emmrich M.	104465
Brown D.	104460
Ebelthite C.	104456
Ryderstedt M.	104453
Henry R.	104444
Davletshin V.	104439
Dancevic F.	104433
Soeda G.	104424
Soderling R.	104417
Fleming C.	104415
Recouderc L.	104397
Burquier G.	104393
Ferreiro F.	104392
Tipsarevic J.	104386
Poerschke L.	104385
Rocha D.	104383
Reid T.	104379
Bogdanovic A.	104372
Minar I.	104371
Koning M.	104361
Chadaj A.	104360

Cook L.	104355
Machado R.	104349
Ancic M.	104339
Monaco J.	104338
Brizzi A.	104334
Petzschner P.	104332
Ram R.	104330
Darcis S.	104327
Ager J.	104325
Dabul B.	104314
Seppi A.	104312
Tavares L.	104311
Przysieczny M.	104308
Gregorc L.	104305
Durek R.	104295
Lu H.	104294
Jaziri M.	104291
Korteling S.	104287
Condor F.	104284
Deheart R.	104281
Topic J.	104275
Popp T.	104272
Verdasco F.	104269
Falla A.	104268
Cipolla F.	104262
Sun P.	104261
Kohlschreiber P.	104259
Reix C.	104256
Mayer F.	104252
Chen T.	104251
Amritraj P.	104250
Middelkoop M.	104235
Nielsen F.	104233
Koellerer D.	104230
Burniol G.	104228
Gensse A.	104227
Knittel B.	104226
Hajek J.	104225
Rettenmaier T.	104224
Bohli S.	104219
Stadler S.	104217
Gonzalez M.	104216
Andreev I.	104214
Valent R.	104211
Eitzinger R.	104209

Patriarca D.	104208
Cuadrado C.	104196
Vagnozzi S.	104195
Crugnola M.	104192
Tuksar S.	104182
Ionita V.	104181
Muller G.	104180
Echagaray B.	104179
Warburg S.	104173
Bloomfield R.	104171
Widom T.	104167
Peric T.	104161
Dlouhy L.	104160
Anderson A.	104158
Capdeville P.	104154
Cruciat A.	104149
Olejniczak D.	104140
Gonzalez S.	104136
Kennedy A.	104126
Berlocq C.	104122
Summerer T.	104121
Lapentti G.	104117
Kimmich M.	104115
Salamanca C.	104113
Balleret B.	104112
Walter L.	104106
Tursunov D.	104098
Guez D.	104095
Kondo H.	104082
Pashanski B.	104081
Acasuso J.	104076
Klaasen R.	104071
Witten J.	104070
Ginepri R.	104068
Norby R.	104059
Karanusic R.	104056
Roddick A.	104053
Lopez M.	104043
Harboe P.	104037
Rea S.	104036
Reynolds B.	104035
Vlasov D.	104034
Johansson J.	104026
Delic A.	104025
Renavand N.	104023

Youzhny M.	104022
Mirnegg M.	104021
Vliegen K.	104019
Childs L.	104014
Kindlmann D.	104010
Ghem A.	104007
Prpic F.	104004
Crivoi V.	104003
Wilson B.	104001
Alves T.	103999
Kubot L.	103997
Robredo T.	103990
Lukaev R.	103986
Chaki R.	103978
Nunez E.	103975
Beck K.	103971
Ferrer D.	103970
Borvanov R.	103969
Lammer M.	103967
Klein T.	103964
Urban F.	103963
Abel M.	103948
Moraru G.	103925
Nijaki R.	103923
Ratiwatana S.	103920
Mahut N.	103917
Awadhy O.	103914
Flanagan I.	103913
Radic M.	103912
Coria G.	103909
Mertl J.	103902
Nalbandian D.	103900
Benneteau J.	103898
King P.	103894
Lorenzi P.	103893
Fish M.	103888
Dailey Z.	103878
Elgin M.	103862
Kunitsyn I.	103857
Lopez F.	103852
Niland C.	103851
Decoud S.	103850
Aldi F.	103849
Bohane A.	103847
Chiudinelli M.	103843

Peralta J.	103841
Perez D.	103837
Volandri F.	103835
Gremelmayr D.	103823
Lipsky S.	103822
Udomchoke D.	103821
Mackin A.	103820
Federer R.	103819
Nieminen J.	103813
Hanescu V.	103812
Starace P.	103808
Gard C.	103805
Prodon E.	103799
Becker B.	103794
Miyao J.	103792
Labadze I.	103789
Davydenko N.	103786
Junaid R.	103782
Melzer J.	103781
Parada F.	103775
Slanar M.	103763
Cakl T.	103761
Dent T.	103758
Hilton M.	103754
Greul S.	103752
Minar J.	103748
Meffert D.	103747
Massa E.	103737
Zovko L.	103735
Murray J.	103730
Serra F.	103722
Hewitt L.	103720
Martin D.	103718
Vico U.	103716
Vinciguerra A.	103714
Levinsky J.	103711
Pless K.	103709
Pospisil J.	103708
Ianni S.	103702
Hartfield D.	103701
Hemmes F.	103699
Rochus O.	103694
Ascione T.	103693
Matkowski M.	103690
Dorsch B.	103684

Yani M.	103676
Junqueira D.	103675
Mello R.	103672
Boeker M.	103663
Klec I.	103657
Montanes A.	103656
Smit J.	103653
Uebel L.	103646
Jiang S.	103642
Horna L.	103632
Tenconi T.	103626
Adaktusson J.	103623
Levine I.	103621
Behrmann M.	103609
Estrella V.	103607
Gonzalez F.	103602
Malisse X.	103598
Marach O.	103594
Haehnel J.	103592
Quintero M.	103590
Fyrstenberg M.	103586
Berrer M.	103582
Saretta F.	103581
Peya A.	103580
Llodra M.	103566
Robert S.	103565
Hippensteel K.	103560
Sherwood D.	103559
Jones A.	103553
Arquez P.	103546
Ram A.	103544
Marrero D.	103543
Auckland J.	103538
Patience O.	103535
Devilder N.	103534
Fleishman Z.	103530
Jeanpierre J.	103527
Miranda I.	103525
Bopanna R.	103523
Stoppini A.	103521
Snobel P.	103520
Healey N.	103516
Steckley R.	103510
Mazarakis V.	103505
Vik R.	103503

Safin M.	103498
Derepasko A.	103497
Ventura S.	103490
Luzzi F.	103487
Blake J.	103484
Veronelli D.	103478
Maamoun K.	103472
Kendrick R.	103470
Mankad H.	103468
Logtens P.	103465
Kralert P.	103457
Mertinak M.	103455
Massu N.	103454
Phau B.	103451
Piccari F.	103437
Luczak P.	103429
Clemens T.	103424
Fitz S.	103422
Tabara M.	103420
Vahaly B.	103408
Terachi T.	103403
Hernych J.	103401
Simoni A.	103397
Silva J.	103395
Schukin Y.	103393
Srichaphan P.	103387
Baccanello P.	103385
Roitman S.	103373
Agaev E.	103371
Ayala D.	103363
Todero N.	103358
Kratochvil M.	103356
Neumuller I.	103353
Golovanov D.	103350
Ljubicic I.	103344
Whitehouse W.	103339
Karlovic I.	103333
Pedroso A.	103328
Moodie W.	103325
Morrison J.	103319
Krajan Z.	103318
Elsner D.	103305
Kim A.	103297
Traykov I.	103296
Rochus C.	103294



Gaudio G.	103292
Zitko R.	103290
Pfeiffer K.	103287
Stepanek R.	103285
Sousa T.	103282
Svarc L.	103281
Verkerk M.	103280
Capalik K.	103270
Puerta M.	103264
Vajda I.	103256
Allegro Y.	103254
Cadart R.	103250
Levy H.	103248
Giner J.	103242
Kim K.	103240
Sarstrand M.	103239
Iwami T.	103230
Daniel M.	103228
Larose S.	103225
Azzaro L.	103209
Grosjean S.	103206
Armando H.	103204
Serrano O.	103202
Garcia A.	103201
Bower J.	103198
Wessels P.	103193
Russell M.	103188
Bryan M.	103185
Bryan B.	103184
Hipfl M.	103182
Vanek J.	103181
Settergren J.	103178
Sanchez D.	103176
Matsui T.	103175
Okun N.	103174
Sluiter R.	103171
Hernandez O.	103169
Voltchkov V.	103166
Fraile G.	103165
Haas T.	103163
Petrovic D.	103162
Parmar A.	103157
Crabb J.	103154
Elseneer G.	103153
Zabaleta M.	103151

Hellstrom M.	103149
Hadad A.	103145
Mutis O.	103133
Onoda M.	103131
Moldovan I.	103118
Lee M.	103114
Bracciali D.	103110
Ouahabi T.	103109
Rehnquist B.	103104
Hrbaty D.	103103
Hui J.	103102
Spracklin T.	103097
Clement A.	103096
Matias N.	103092
Canas G.	103084
Hantschk M.	103082
Hanley P.	103080
Economidis K.	103077
Belal C.	103068
Nielsen M.	103066
Hernandez A.	103060
Delfino M.	103052
Varlet J.	103047
Phillips T.	103039
Carlson A.	103037
Lopes H.	103033
Fogues F.	103032
Tahiri M.	103029
Mirnyi M.	103018
Kiefer N.	103017
Melo D.	103015
Sabau R.	103009
Galvani S.	102999
Gambill J. M.	102998
Gamonah H.	102997
Sa A.	102987
Marcaccio G.	102985
Garcia M.	102984
Andersson H.	102983
Ishii Y.	102982
Coetzee J.	102979
Erlich J.	102968
Gicquel M.	102967
Jonsson F.	102966
Delgado J.	102965

Tkalec M.	102959
Kutsenko V.	102957
Khalfan S.	102956
Vicente F.	102950
Bourgeois L.	102945
Coutelot N.	102934
Gimelstob J.	102925
Fukarek O.	102921
Stoliarov A.	102911
Navarro S.	102910
Koubek S.	102905
Friedl L.	102904
Blake T.	102902
Delgado R.	102887
Cermak F.	102886
Uppal V.	102885
Philippoussis M.	102882
Skoch D.	102881
Popp A.	102880
Smith L.	102875
Blanco G.	102869
Bohaboy D.	102865
Moser F.	102864
Suzuki T.	102863
Calleri A.	102860
Kuerten G.	102856
Brasington J.	102855
Schalken S.	102854
Galimberti G.	102851
Moya C.	102845
Lapentti N.	102839
Wassen R.	102836
Milligan L.	102835
Goldstein P.	102834
Diaz J.	102831
Mamiit C.	102821
Kempes E.	102817
Vemic D.	102811
Pretzsch A.	102809
Flygt K.	102808
Breen M.	102805
Zimonjic N.	102800
Norman M.	102796
Gooding J.	102789
Vacek J.	102788

Weiner G.	102785
Schuettler R.	102783
Niemeyer F.	102780
Ilie A.	102776
Browne F.	102768
Escude N.	102765
Perry T.	102751
Ellwood B.	102743
Kauffmann C.	102735
Heuberger I.	102730
Savolt A.	102722
Zib T.	102720
Hirigoyen I.	102715
Rios M.	102701
Burgsmuller L.	102694
Olguin L.	102693
Wiltchnig H.	102687
Behr N.	102669
Iwabuchi S.	102667
Vinck C.	102652
Carretero R.	102650
Squillari F.	102644
Saulnier C.	102642
Saoudi S.	102630
Burrieza O.	102629
Tiilikainen K.	102617
Costa A.	102610
Balcells J.	102607
Koll H.	102595
Prieto S.	102585
Trifu G.	102574
Carraz G.	102571
Bastl G.	102567
Waske A.	102565
Johansson T.	102563
Novak J.	102562
Taino E.	102558
Willman D.	102557
Moyano D.	102554
Braga P.	102545
Ulihrach B.	102539
Ketola T.	102538
Knippschild J.	102533
Shimada T.	102529
Godwin N.	102524

Dreekmann H.	102521
Malcor O.	102520
Sirianni J.	102512
Harsanyi P.	102502
Washington M.	102497
Behrend T.	102494
Etlis G.	102477
Schiessling T.	102473
Craca M.	102472
Peppas S.	102469
Syed F.	102467
Kroslak J.	102466
Larkham T.	102464
Thomas J.	102458
MacLagan M.	102457
Mantilla F.	102456
Manta L.	102455
Henman T.	102450
Medvedev A.	102446
Cowan B.	102443
Voinea A.	102437
Spadea V.	102434
Navarra M.	102433
Dupre T.	102406
Bhupathi M.	102405
Draper S.	102401
Knowle J.	102382
Eschauer W.	102381
Pescariu D.	102375
Corretja A.	102374
Boutter J.	102369
Enqvist T.	102358
Tomashevich D.	102347
Kucera K.	102344
Kafelnikov Y.	102338
Pavel A.	102318
Kohlmann M.	102306
Motomura G.	102300
Portas A.	102287
Black W.	102286
Chaen T.	102282
Arazi H.	102271
Salzenstein J.	102269
Golmard J.	102259
Rusedski G.	102257

Hood M.	102251
Couto E.	102250
Gaudenzi A.	102247
Charpentier M.	102243
Costa F.	102239
Berasategui A.	102238
Gross O.	102234
Paes L.	102233
Calatrava A.	102231
Sargsian S.	102227
Alami K.	102223
Eagle J.	102215
Weinzierl J.	102207
Lareau S.	102205
Carlsen K.	102202
Roux L.	102201
Prinosil D.	102184
Dupuis A.	102179
Joyce M.	102169
Woodruff C.	102162
Rafter P.	102158
Ondruska M.	102154
Puentes G.	102153
Santoro F.	102148
Thomann N.	102143
Nestor D.	102110
Sanguinetti D.	102106
Dabrowski B.	102100
Srichaphan N.	102097
Damm M.	102093
Ogorodov O.	102087
Shvec A.	102077
Bjorkman J.	102035
Gumy H.	102024
Luxa P.	102023
Chang M.	102021
Martelli M.	101997
Hadad M.	101991
Krajicek R.	101990
Bozic I.	101970
Stanoytchev O.	101969
Ferreira W.	101965
Ivanisevic G.	101964
Sampras P.	101948
Santopadre V.	101947

Mota B.	101938
Koenig R.	101937
Pescosolido S.	101924
Huet S.	101902
Meligeni F.	101897
Stark J.	101890
Woodbridge T.	101889
Vacek D.	101888
Arthurs W.	101885
Reichel A.	101871
Norman D.	101868
Bruguera S.	101843
Tebbutt M.	101833
Rosset M.	101820
Dosedel S.	101792
Martin T.	101774
Oncins J.	101761
Caratti C.	101750
Furlan R.	101746
Agassi A.	101736
Fromberg R.	101735
Siemerink J.	101733
Stoltenberg J.	101727
Larsson M.	101723
Rodriguez M.	101675
Black B.	101647
Pioline C.	101611
Tarango J.	101543
Clavet F.	101532
Tobon M.	101495
Muster T.	101404
Gustafsson M.	101320
Pozzi G.	101150
Agenor R.	101086
Zverev A.	100644

### DE-PARA de Rodadas (Rounds)

draw_size	round	round_odds
4	RR	Round Robin
8	RR	Round Robin
8	SF	Semifinals
8	F	The Final

32	R32	1st Round
32	R16	2nd Round
32	QF	Quarterfinals
32	SF	Semifinals
32	F	The Final
48	R64	1st Round
48	R32	2nd Round
48	R16	3rd Round
48	QF	Quarterfinals
48	SF	Semifinals
48	F	The Final
56	R64	1st Round
56	R32	2nd Round
56	R16	3rd Round
56	QF	Quarterfinals
56	SF	Semifinals
56	F	The Final
64	R64	1st Round
64	R32	2nd Round
64	R16	3rd Round
64	QF	Quarterfinals
64	SF	Semifinals
64	F	The Final
64	RR	Round Robin
96	R128	1st Round
96	R64	2nd Round
96	R32	3rd Round
96	R16	4th Round
96	QF	Quarterfinals
96	SF	Semifinals
96	F	The Final
128	R128	1st Round
128	R64	2nd Round
128	R32	3rd Round
128	R16	4th Round
128	QF	Quarterfinals
128	SF	Semifinals
128	F	The Final