Starting team selection

This first document looks at how to decide on an ***initial team*** to begin the season with. There are numerous factors which must be considered.

If we can maximise ROI we maximize total points throughout the season.

Man United Man City Burnley Villa do not play first week

Therefore, the primary objective is **maximise ROI across 15 players who play week in week out and rotate players based on fixtures**

This is constrained by finding the best starting 11 that do not include players from any of these 4 teams.

For the first gameweek, I'm just going to ignore all new players and stick with well estalished players to begin with.

In [1]:

```
#package import

import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
```

In [214]:

```
#data import

#basic data
data = pd.read_csv("https://raw.githubusercontent.com/vaastav/Fantasy-Premier-League/master/data/2020-21/cleaned_players.csv")
data["ROI"] = data["total_points"]/data["now_cost"]
data["full_name"] = data["first_name"] + " " + data["second_name"]


#raw_data
raw_data = pd.read_csv("https://raw.githubusercontent.com/vaastav/Fantasy-Premier-League/master/data/2020-21/players_raw.csv")
# 1= gk ... 4 = FWD (element type)


#player ID
player_id = pd.read_csv("https://raw.githubusercontent.com/vaastav/Fantasy-Premier-League/master/data/2020-21/player_idlist.csv")

#raw_data[["id", "now_cost"]].head(10)
player_id["full_name"] = player_id["full_name"] = player_id["first_name"] + " " + player_id["second_name"]


data["id"] = player_id["id"]
data["position"] = raw_data["element_type"]
#data.set_index("full_name", inplace = False, drop = False)

star_team = [] #list of star team (by index) # creating list of star players to be appended

data.head()
```

Out[214]:

	first_name	second_name	goals_scored	assists	total_points	minutes	goals_conced
0	Mesut	Özil	1	3	53	1439	
1	Sokratis	Papastathopoulos	2	0	57	1696	
2	David	Luiz Moreira Marinho	2	1	94	2809	
3	Pierre-Emerick	Aubameyang	22	5	205	3136	
4	Cédric	Soares	1	1	61	1553	

5 rows × 22 columns

First five rows of the dataset

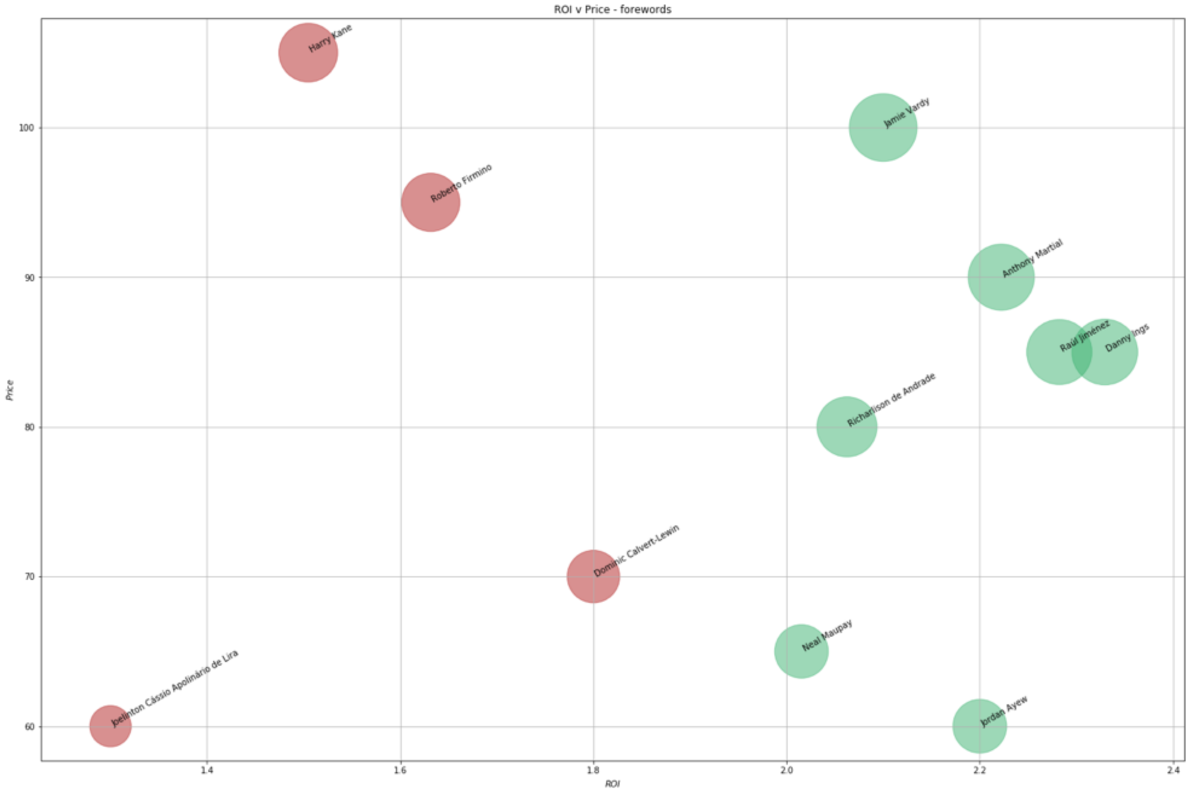
Existing teams and players

This is an interactive function which can find the best players in positions for certain prices. Price is on the Y axis, so the further they are up the graph the more expensive they are. ROI is on the x axis so the further right they are on the graph the better they are in terms of ROI. The size of the circle represents how many total points they have.

In [466]:

```
#Calcualting and plotting ROI against price

#This tool can be used as a quick visual tool to explore players by input


def graph_roi(data, position, min_minutes, minROI, idealROI, max_cost):
    data = data[data.minutes > min_minutes].reset_index(drop = True)
    data = data[data.position == position].reset_index(drop = True)
    data = data[data.ROI > minROI].reset_index(drop = True)
    data = data[data.now_cost <= max_cost].reset_index(drop = True)

    positions = ["inplace", "goalkeeper", "defender", "midfielder", "foreword"]
    player_names = data["full_name"]
    x = data["ROI"]
    y = data["now_cost"]
    fig, ax = plt.subplots(figsize = (24,16))

    for i, txt in enumerate(player_names):
        ax.annotate(txt, (x[i], y[i]), rotation = 30)


    ax.grid(which = "major")
    ax.scatter(data["ROI"], data["now_cost"], s = data["total_points"]*30, alpha = 0.5,
                color = np.where(data.ROI < idealROI, "firebrick", "mediumseagreen" ))
    ax.set_xlabel("$ROI$")
    ax.set_ylabel("$Price$")
    ax.set_title("ROI v Price - {}s".format(positions[position]))

    return "the graph below shows the ROI v price of {}s with a minimum minutes of {}".format(
        positions[position], min_minutes)

# GK = 1, DEF = 2, MID = 3, FWD = 4
#position, min_minutes, min_roi, ideal_roi, min_cost

#By inputting players position, minimum minutes, minimum ROI, Ideal ROI, and maximum cost
graph_roi(data, 4, 2500, 0, 2, 120)
```

```
Out[466]: 'the graph below shows the ROI v price of forewords with a minimum
          minutes of 2500'
```



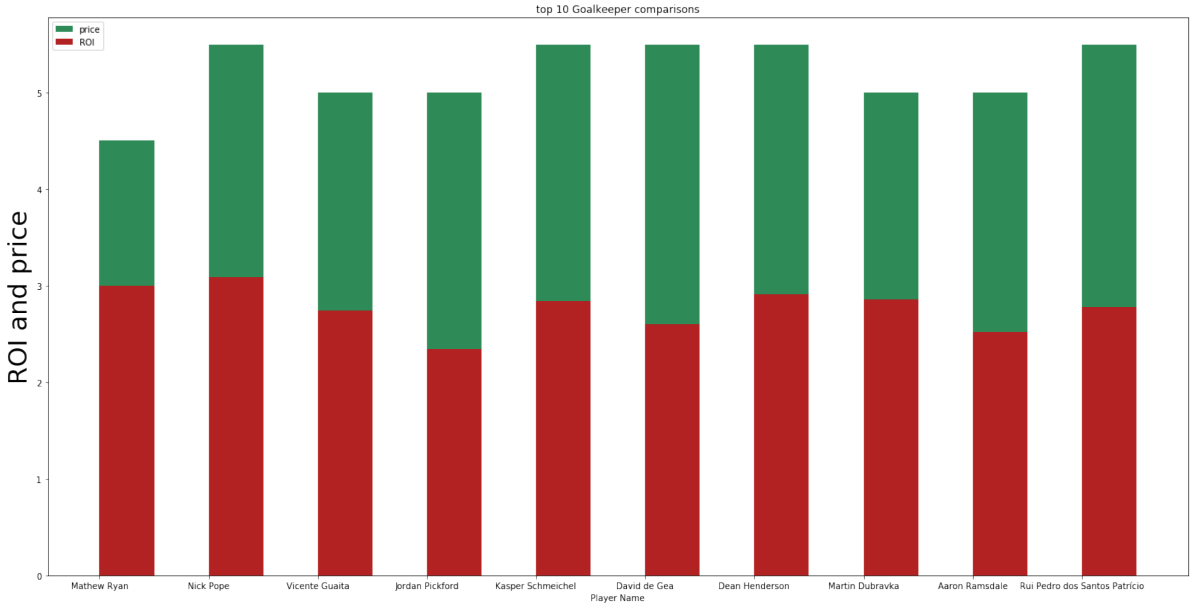
Goalkeepers

In [472]:

```python
GK = data[data.position == 1]
GK = GK.loc[:, ["full_name", "ROI", "now_cost", "minutes"]]
top10GK = list(GK.ROI.nlargest(10).index)
GK = GK[GK.index.isin(top10GK)]


#plt.plot(x = GK["full_name"], y = GK["ROI"])
fig, ax = plt.subplots(figsize = (24,12))

ax.bar(GK["full_name"], GK["now_cost"]/10, 0.5, align = "edge", color = "seagreen")


ax.bar(GK["full_name"], GK["ROI"], 0.5, align = "edge", color = "firebrick")
#ax.bar(GK["full_name"], GK["now_cost"]/10, 0.5, align = "edge", color = "seagreen")

ax.set_xlabel("Player Name")
ax.set_ylabel("ROI and price", size = 30)
ax.set_title("top 10 Goalkeeper comparisons")
ax.legend(["price", "ROI"])

#From looking at this, it suggests that nick pope is still the most favourable choice.
#A nice insight is looking at Matt Ryan, he is significantly cheaper than the rest at 4.5m and has the second highest
#ROI (this is obviously effected by his low price). However, he is a perfect choice for backup - low price, always plays, high ROI.
```

Out[472]:

```
<matplotlib.legend.Legend at 0x7f91d96870d0>
```



From looking at this, it suggests that nick pope is still the most favourable choice. A nice insight is looking at Matt Ryan, he is significantly cheaper than the rest at 4.5m and has the second highest ROI (this is obviously effected by his low price). However, he is a perfect choice for backup - low price, always plays, high ROI. Therfore, ***Nick Pope stays first choice with Matt Ryan as back up.***

Defenders

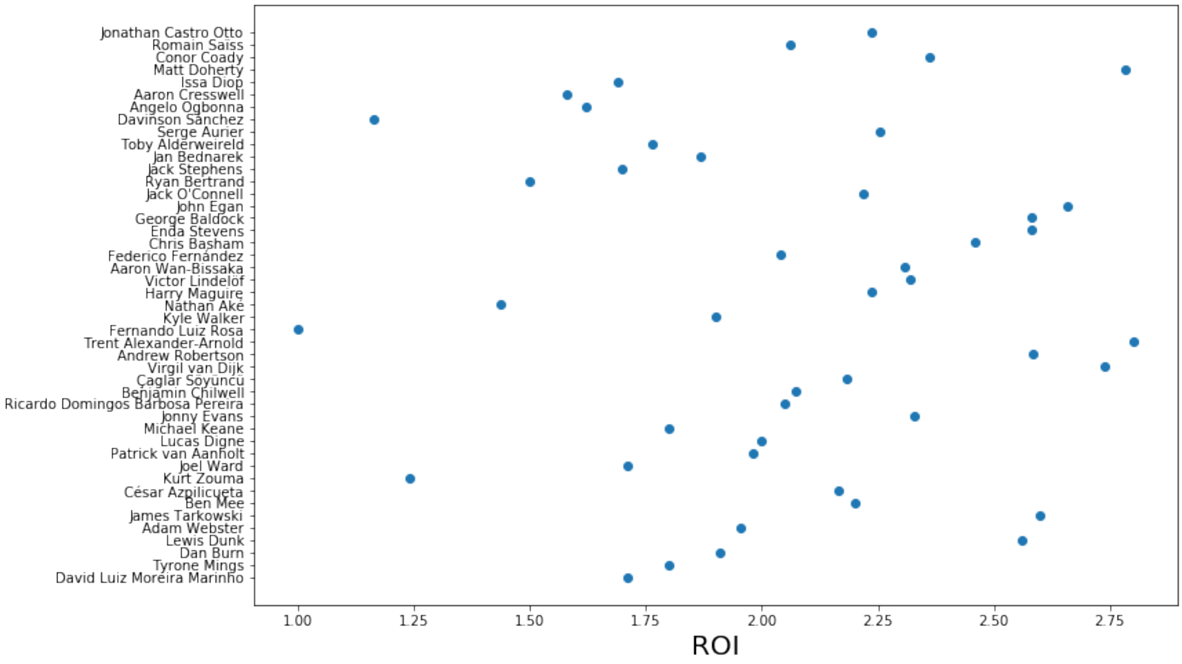
Defenders will be significantly more difficult than goalkeepers. We have to chose 5 defenders. As mentioned before we have a criteria of they must play week in week out and have the highest ROI in that category. It wasnt neccessary with goalkeepers as they tend to play most of the season and rarely get injured (of course leno did but arsenals defence is shocking so he isnt getting anywhere near the team anyway). With defenders however, sometimes they can be injured and players can be brought in in January also. So we can look at ROM (return on minutes) to compliment ROI here.

In [473]:

```
#First of all we can look at players who have averaged above 60 minutes a game all season and look at their highest ROI's.
# 60 * 38 = 2280 (rounding up to 2300 for convenience)
DEF = data[data.position == 2]
DEF = DEF[DEF.minutes >= 2300] # this leaves 45 players, more than enough to chose from.

fig, ax = plt.subplots(figsize = (12,8))
ax.scatter(DEF.ROI, DEF.full_name)
ax.set_xlabel("ROI", size = 20)
```

Out[473]:

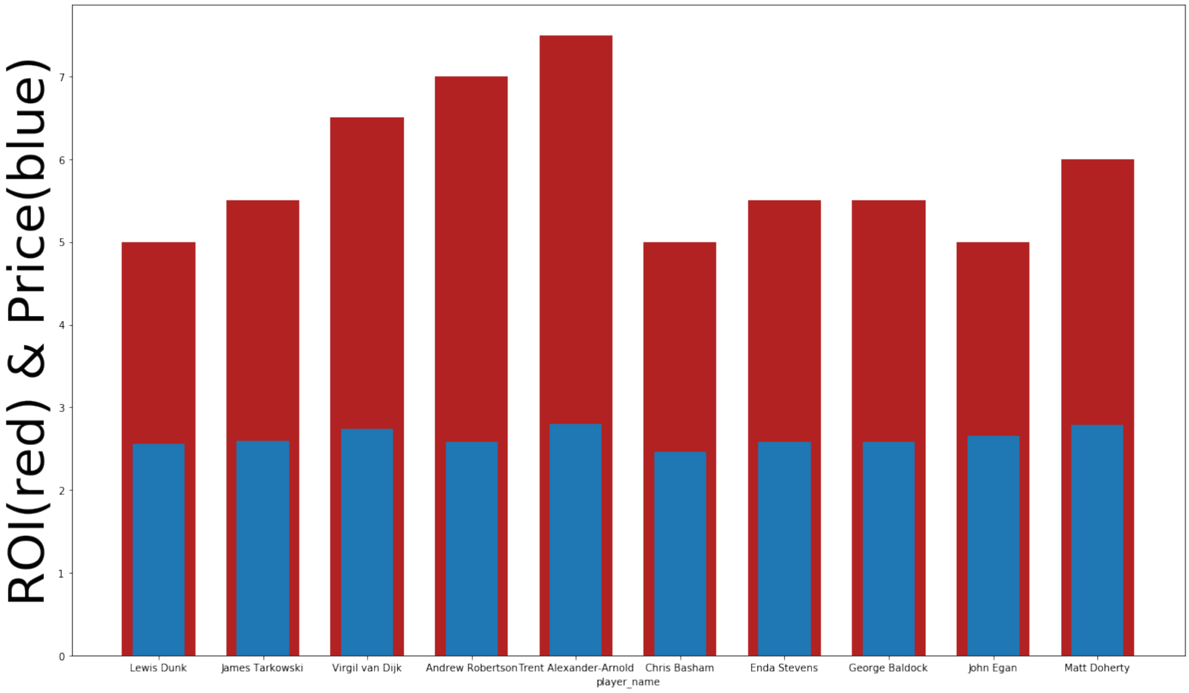
```
Text(0.5, 0, 'ROI')
```



From this we can see that even with an ROI of 2.25 there are still plenty of players to chose from. Therefore, we can filter for this and plot again.

```
In [470]: DEF = DEF[DEF.ROI > 2.4]

          fig, ax = plt.subplots(figsize = (20,12))
          ax.bar(DEF.full_name, DEF.now_cost/10, width = 0.7, color = "firebr
          ick")
          ax.bar(DEF.full_name, DEF.ROI, width = 0.5)
          ax.set_xlabel("player_name")
          ax.set_ylabel("ROI(red) & Price(blue)", size = 50)
```

```
Out[470]: Text(0, 0.5, 'ROI(red) & Price(blue)')
```



As can be seen there is a correlation between highest ROI and price, suggesting that the top players are still undervalued. It therefore follows, that we need to strike a balance between the highest ROI players as well as the cheapest players with the highest ROI.

A potential solution is having trent along with dunk, basham, egan and tarkowski. This is quite a diverse range of teams, only basham and egan are from the same team. This would be a total cost of 28m. Plus the 10m for the goalkeepers. This would leave 62 million for the remaining 8 players (averaging 7.75 per player from the initial 6.67)This would therefore be an attack focused team.

Another solution would be to also include van dijk and replace one of the sheffield players leaving a total of 30m (this would leave 7.5 per player).

It also looks likely that man city defense will improve if they spend considerable money on it. Nathan Ake is also a potential option at 5.5m. But it might be better to wait and see with him - he may not start.

From looking at the fxtures, it can also be observed that if you rotate Wolves and Burnley defenders (two notoriously defensive teams) you would also only play the top 6 teams twice out of 36 weeks.

Come back to defence.

Star players

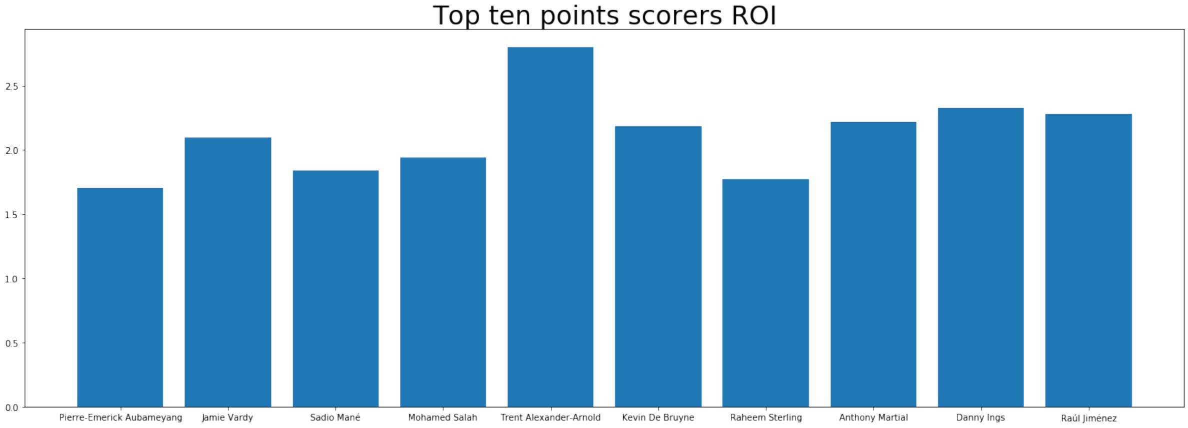
The one thing that lets down the ROI strategy is the inclusion of double points captains (and vice captains). We therefore, need to identify at least two players who are extremely high points scorers who will be the captian choice each week. Therefore, the approach is to chose the players with the highest ROI from the top scoring players.

In [474]:

```
top_points = list(data.total_points.nlargest(10).index)
top_players = data.index.isin(top_points)
top_players = data[top_players]

fig, ax = plt.subplots(figsize = (24,8))
ax.bar(top_players.full_name, top_players.ROI)
ax.set_title("Top ten points scorers ROI", size = 30)
```

Out[474]:

```
Text(0.5, 1.0, 'Top ten points scorers ROI')
```

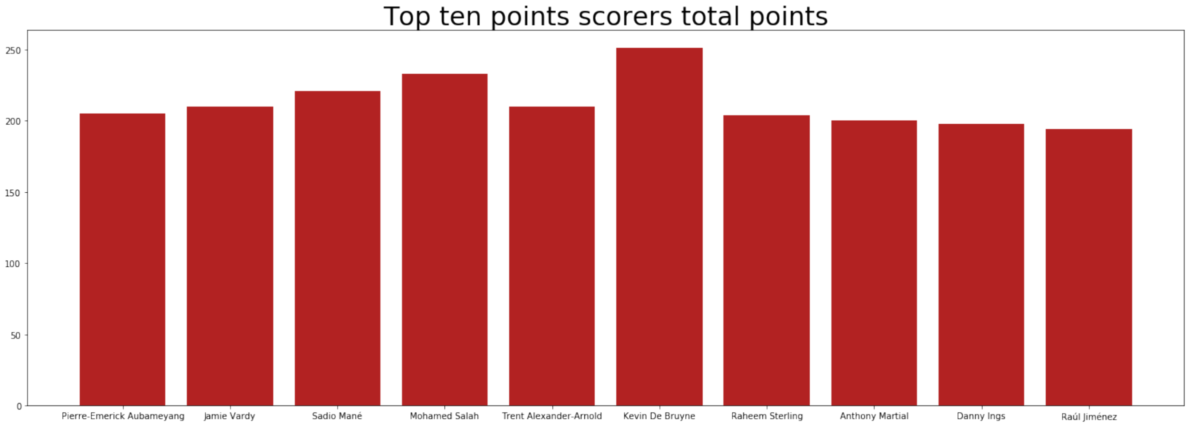


In [476]:

```
fig, ax = plt.subplots(figsize = (24,8))
ax.bar(top_players.full_name, top_players.total_points, color = "firebrick")
ax.set_title("Top ten points scorers total points", size = 30)
```

Out[476]:

```
Text(0.5, 1.0, 'Top ten points scorers total points')
```



In [206]:

```
MIDFWD = data[data.position.isin([3,4])]
top_roi = list(MIDFWD.ROI.nlargest(10).index)
top_roi = data.index.isin(top_roi)
top_roi = data[top_roi]
top_roi_index = list(top_roi.index)
top_roi_index

top_players_index = list(top_players.index)
top_players_index

high_roi_points = [x for x in top_players_index if x in top_roi_index]
high_roi_points

high_roi_points = data.index.isin(high_roi_points)
data[high_roi_points]

#PLayers who are in both the top 10 for ROI and points scored can be seen below.
```

Out[206]:

	first_name	second_name	goals_scored	assists	total_points	minutes	goals_concede
272	Kevin	De Bruyne	13	23	251	2790	2
303	Anthony	Martial	17	9	200	2625	2
369	Danny	Ings	22	2	198	2800	4
466	Raúl	Jiménez	17	7	194	3241	3

4 rows × 22 columns

As can be seen above, only these four players are in the top 10 for points scored and ROI. The graphs above this show the ROI of top points scorers, but only these 4 players are also in the top 10 ROI. Therefore, as can be seen it looks like KDB and Anthony martial are the players to go with. This does not mean that ings and and jiminez are bad choices - they could very likely still make the team. KDB is a dead cert, he is top points scorer and has a really high ROI. Any of these options were viable choices, however, using personal insight I think Martial may have the best season (Man united bias). If he had bruno fernandes behind him the full season his points total would have been much higher. He was also injured for 2 months which would have reduced his output.

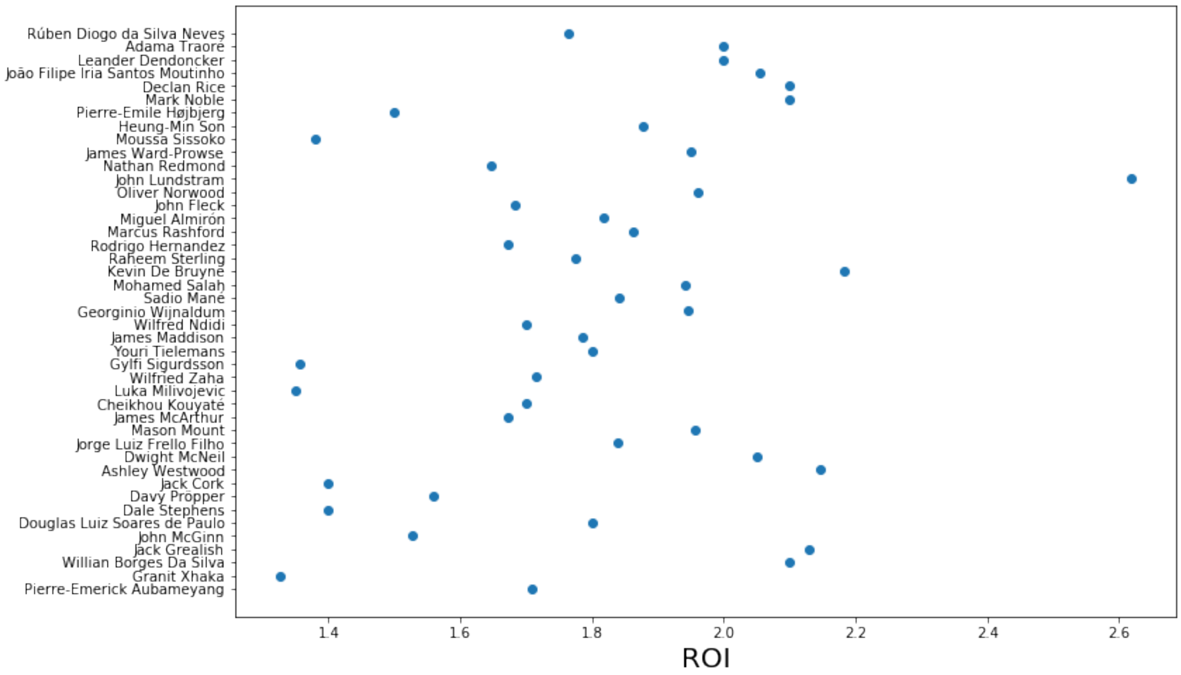
selecting players who are in both of these lists attempts to find the players who are still going to score lots of points but are not overpriced. Players like salah for example score a lot of points but are overpriced. KDB scored more points and even after a price adjustment is less expensive.

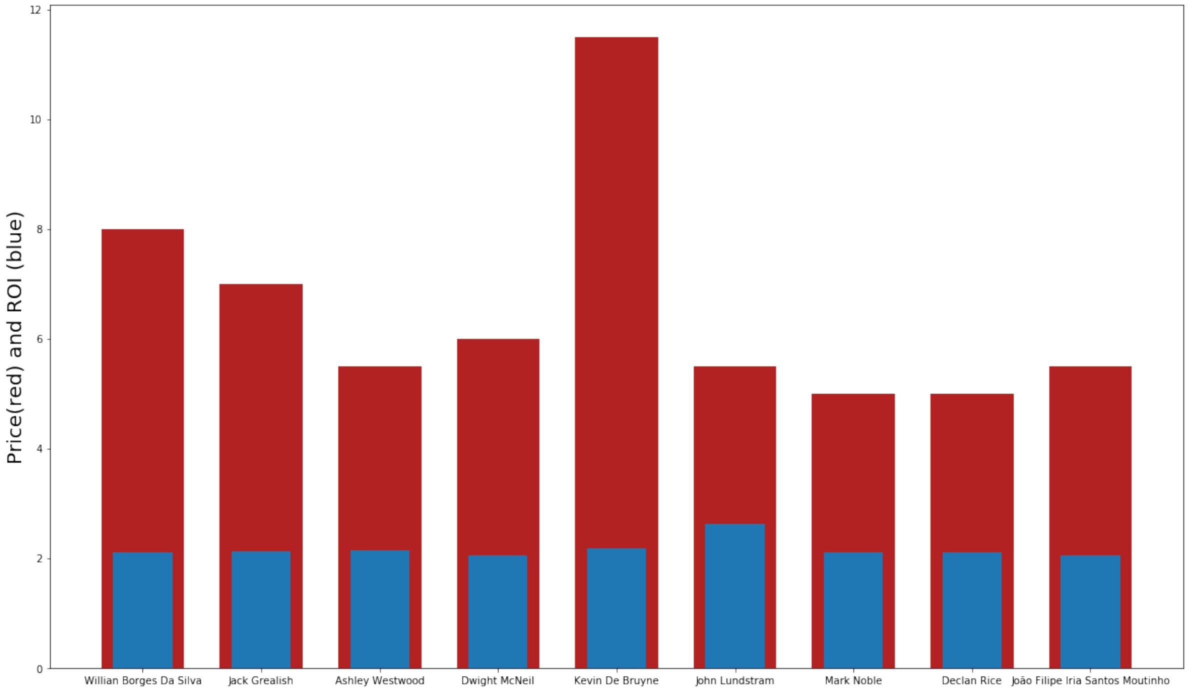
Midfeilders

In [479]:

```
#We can continue on the same logic, by filtering out players who have less than 2300 minutes I am aware this
#will exlcude certain players that have arrived in Jan or been injured - but will come back to this later)

MID = data[data.position == 3]
MID = MID[MID.minutes >= 2300] # this leaves 45 players, more than enough to chose from.

fig, ax = plt.subplots(figsize = (12,8))
ax.scatter(MID.ROI, MID.full_name)
ax.set_xlabel("ROI", size = 20)

#as can be seen again, a cut off of around 1.9 will leave us plenty of options.

MID = MID[MID.ROI > 2]

fig, ax = plt.subplots(figsize = (20,12))
ax.bar(MID.full_name, MID.now_cost/10, width = 0.7, color = "firebrick")
ax.bar(MID.full_name, MID.ROI, width = 0.5)
ax.set_ylabel("Price(red) and ROI (blue)", size = 20)
```

```
Out[479]: Text(0, 0.5, 'Price(red) and ROI (blue)')
```





The top ten midfileders in terms of ROI can be seen above. Of course KDB is there but there are also some surprising faces. There is also a few we can eliminate however due to change in clubs/positions. Lundtram will lose all defensive points therefore he is not a good choice, and Willians move to Arsenal makes him a less promising choice (as a lot of his points came from penalties at Chelsea and he may not take them at arsenal).

Grelish, westwood, mcneil are also not availble week 1 so should not be included. Declan rice, Noble and Moutinho are all therefore viable options with high ROI and relatively low prices.

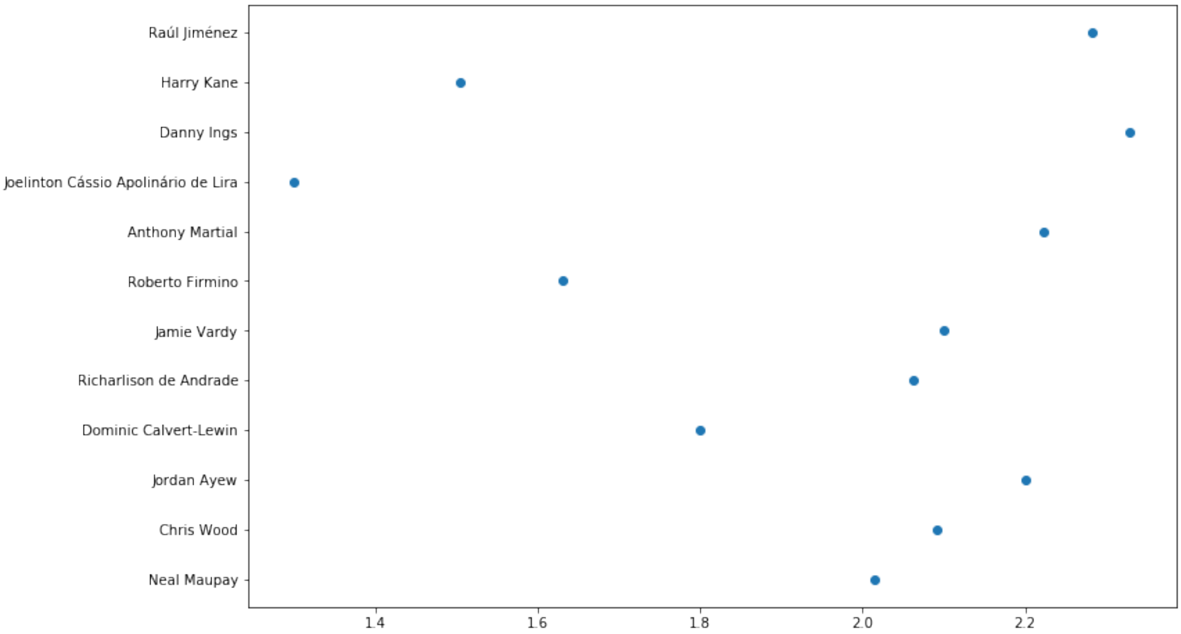
However, these options are also relatively low value. And both the two star players we have included so far will not play first week we should make some adjustments. We can leave KDB out first week and replace him with another star player. Looking back to the list of Star players, Salah is the next highest points scorer and is also playing this week so he is a good choice. He, however, is expensive. If he was to be balanced with the cheaper options mentioned it could work.

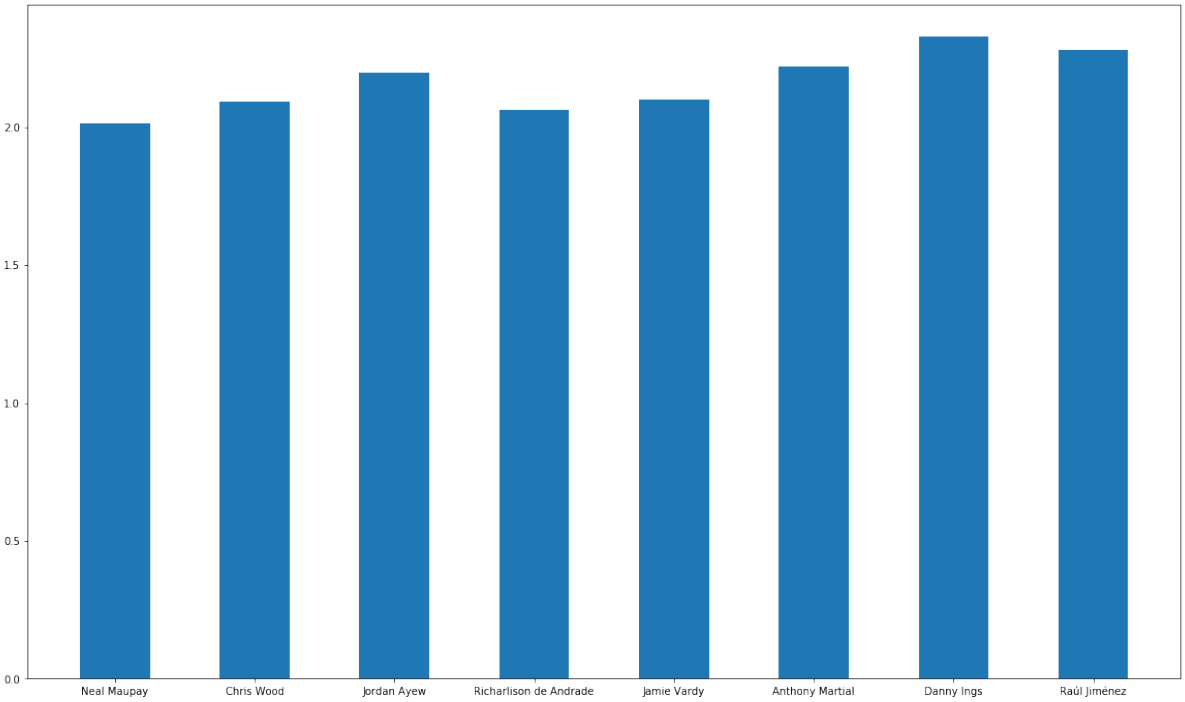
Forwords

In [230]:

```
#A similar methodology can be looked at for forwards:

FWD = data[data.position == 4]
FWD = FWD[FWD.minutes >= 2300] # this leaves 45 players, more than enough to chose from.

fig, ax = plt.subplots(figsize = (12,8))
ax.scatter(FWD.ROI, FWD.full_name)

#as can be seen again, a cut off of around 2 will leave us plenty of options.

FWD = FWD[FWD.ROI > 2]

fig, ax = plt.subplots(figsize = (20,12))
#ax.bar(FWD.full_name, FWD.now_cost/10, width = 0.7, color = "firebrick")
ax.bar(FWD.full_name, FWD.ROI, width = 0.5)
```

```
Out[230]: <BarContainer object of 8 artists>
```





As can be seen above, Danny ings has the highest ROI of any player and is therefore a good choice. This is closely followed by Jiminez. Within the budget contraint I think it is only feasable to have one of these players in. This is followed by Anthony Martial who is already in the team. Ayew is the next highest ROI and also at a significantly lower price and is therefore a solid choice.

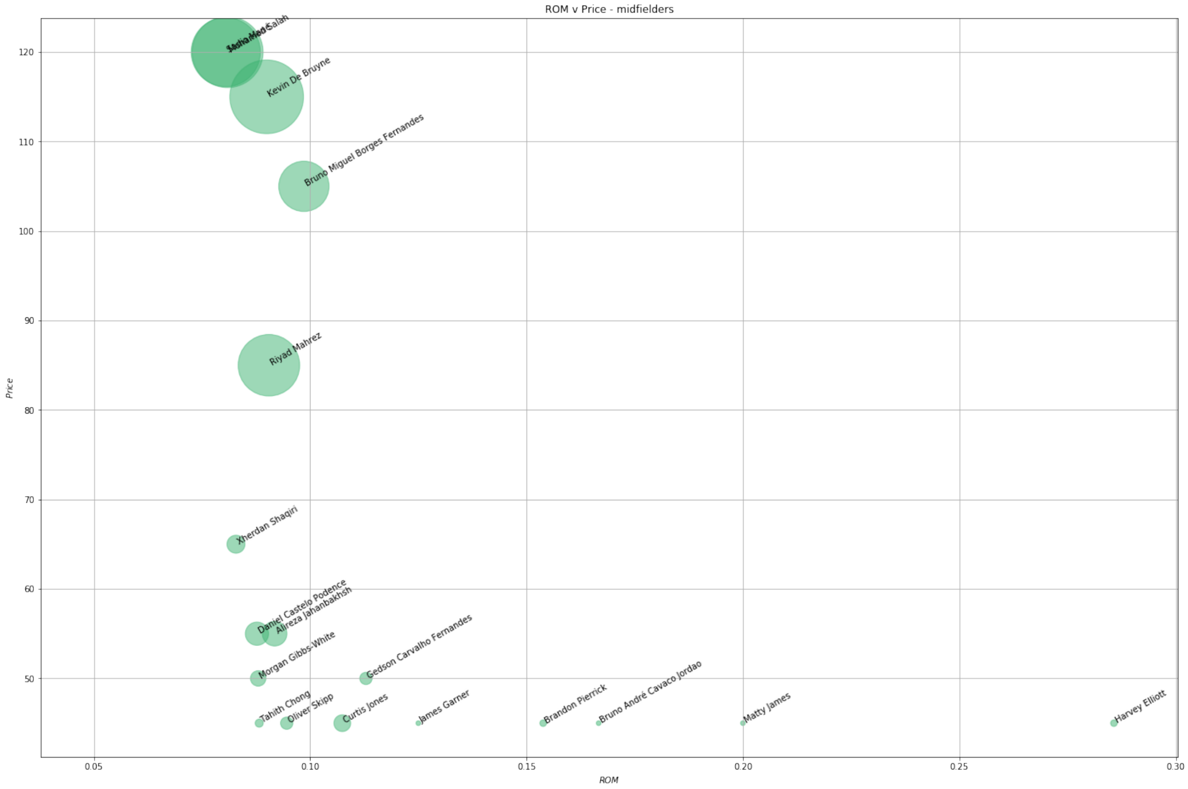
Hidden gems

The ROI model will occasionally look over players who have not played the full/or close to full season. Therefore, we can also look at ROM (return on minutes) which will show how well a player has done with the minutes they have been given.

In [490]:

```
data["ROM"] = data["total_points"]/data["minutes"]


def graph_rom(data, position, selected, minROM, max_cost):
    data = data[data.selected_by_percent < selected].reset_index(drop = True)
    data = data[data.position == position].reset_index(drop = True)
    data = data[data.ROM > minROM].reset_index(drop = True)
    data = data[data.now_cost <= max_cost].reset_index(drop = True)

    positions = ["inplace", "goalkeeper", "defender", "midfielder", "foreword"]
    player_names = data["full_name"]
    x = data["ROM"]
    y = data["now_cost"]
    fig, ax = plt.subplots(figsize = (24,16))

    for i, txt in enumerate(player_names):
        ax.annotate(txt, (x[i], y[i]), rotation = 30)


    ax.grid(which = "major")
    ax.scatter(data["ROM"], data["now_cost"], s = data["total_points"]*30, alpha = 0.5,
                color = "mediumseagreen" )
    ax.set_xlabel("$ROM$")
    ax.set_ylabel("$Price$")
    ax.set_title("ROM v Price - {}s".format(positions[position]))

    return "the graph below shows the ROM v price of {}s".format(position)
```

```
In [494]: graph_rom(data, 3, 50, 0.08, 120)

Out[494]: 'the graph below shows the ROM v price of 3s'
```



Some of these results must be taken with a punch of salt as they have only played a minute amount of minutes. For exmaple, James garner, harvey elliot etc have all played less than 2 games. Curtis Jones has played 6 games and has a decent return in those games. He is unlikely to start going forward though. The two main outstanding players here are Bruno Fernandes and Mahrez. Both players have high ROM and total points(represented by the size of their circle.)With the departure of Sane and David Silva, Mahrez is likely to get more minutes this season. And Bruno fernandes is likely to start almost every game as he is critical to uniteds team. Therefore, both of these players are good options.

Summary

Therefore, considering the above the choices are plentiful. It seems like Pope and Ryan are the best options for Goalkeeper, with Pope starting the majority of the games and bringing Ryan in when Burnleys expected clean sheeets drop too low.

Defenders, it seems like Trent is a must have and there is a wide variety of supplementary options (Dunk, Tarkowski, VVD, Roberston, Basham, Stevens, Baldock, Egen, Doherty, Coady.

Midfielders, De Bruyne is a clear must have, Bruno Fernandes also looks essential. These can be coupled with some cheaper options to compliment such as Rice, Noble, Moutinho, Graelish, westwood or mcneil.

And forwards, Martial, Ings and Jiminez are the clear fronrunners. Ayew is also a fantastic option considering his price. His ROI is only less than the three previously mentioned.