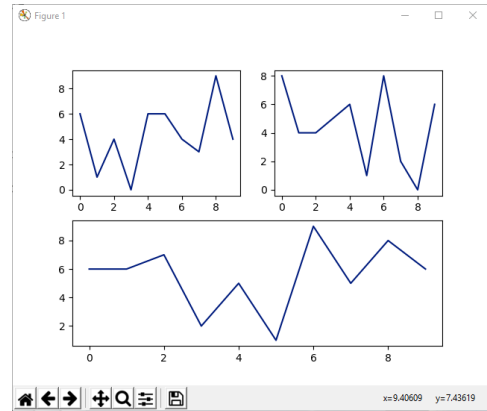


subplots תתי-גרף

ישנם 2 דרכים עיקריות איך להציג כמה גרפים באותו חלון. אך בשביל זה, ניצור פונקציה שתיצור לנו 10 זוגות מספרים באופן רנדומלי הבא:

```
def create_plots():
    xs = []
    ys = []
    for i in range(10):
        x = i
        y = random.randrange(10)
        xs.append(x)
        ys.append(y)
    return xs, ys
```

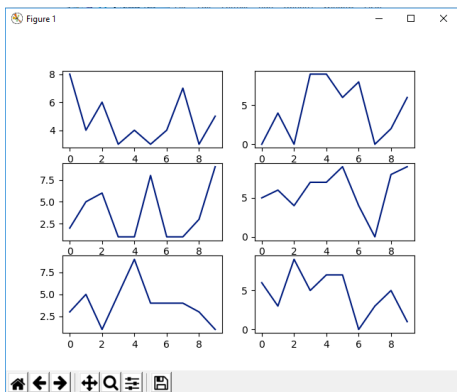


הנה דוגמה מעל השורה הזאת של כמה תתי-גרפים בחלון אחד.
דרך א: הוספה לפי מספור-חלוקה, ועושים זאת כך:

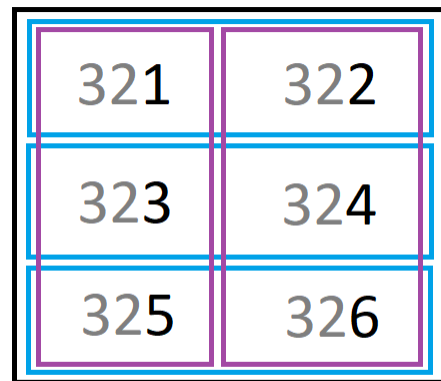
ax1 = fig.add_subplot(221)	ax2 = fig.add_subplot(222)	ax3 = fig.add_subplot(212)
x,y = create_plots()	x,y = create_plots()	x,y = create_plots()
ax1.plot(x,y)	ax2.plot(x,y)	ax3.plot(x,y)

המספור של התתי גרפים מתבצע באופן הבא - יש 3 ספרות:

- הספרה השמאלית מציינת כמה שורות יהיו בחלון הזה
- הספרה האמצעית מציינת כמה עמודות יהיו בחלון הזה
- הספרה הימנית מציינת במספר סידורי למקם את התת גרף שלנו (הנע משמאל לימינה בסדר רץ).



3X2

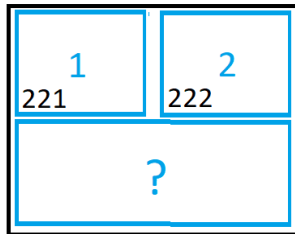


X2X

באופן דיפולטיבי, יש לנו חלון אחד, וזה מתבצע כך:

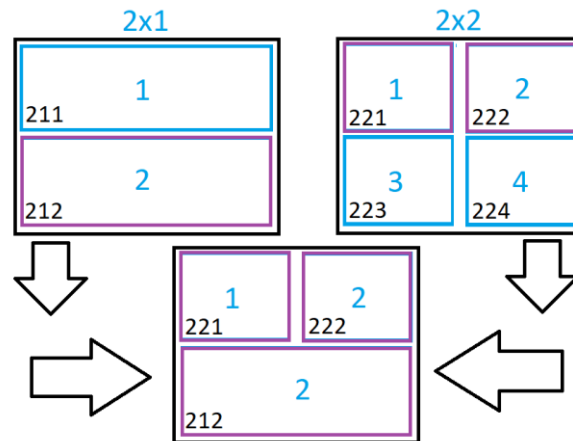
```
ax1 = fig.add_subplot(111)
```

כלומר, חלון בגודל של 1x1, במקום ה-1

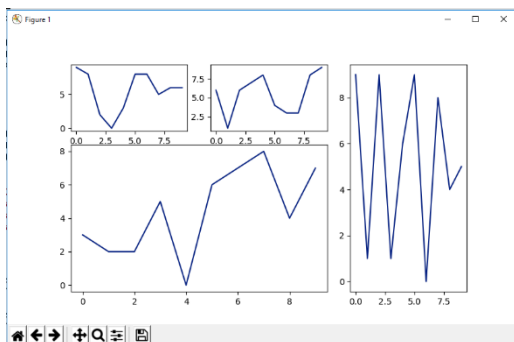


דבר אחרון בדרך הזאת, אני אסביר את הגרף הראשון שהוצג פה בנושא הזה..
 כאשר יש לנו טבלה של 2 שורות (ועמודה אחת) היא מובנת, כאשר יש לנו את טבלה של 2 שורות ו 2 עמודות זה מובן, אבל מה קורה אם יש לנו טבלה כזאת?

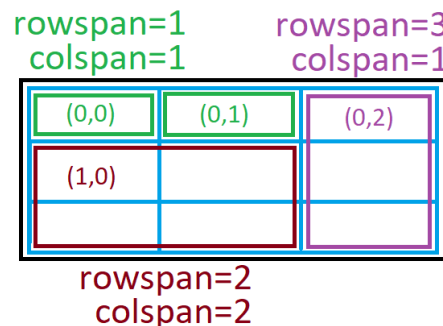
אז מה שקורה פה, זה נוצר לנו 2 טבלאות נפרדות, שאנחנו מתייחסים אליהם אחד על השני, וזה יראה כך:



זרז: היא לבצע זאת בעזרת grid – רשת, אנחנו ניצור רשת בגודל של r, c , ונחליט היכן למקם שמה לפי אינדקס, וניתן לתת לתת-גרף מקום של כמה טבלאות על מקום אחד, למשל זה נראה



כך:



וזה מתבצע בקוד באופן הבא:

`plt.subplot2grid((Rows,Cols), (r_index,c_index), rowspan = AmountR, colspan = AmountC)`

דוגמה:

```
ax = []
ax.append( plt.subplot2grid((3,3), (0,0), rowspan = 1, colspan = 1) )
ax.append( plt.subplot2grid((3,3), (0,1), rowspan = 1, colspan = 1) )
ax.append( plt.subplot2grid((3,3), (0,2), rowspan = 3, colspan = 1) )
ax.append( plt.subplot2grid((3,3), (1,0), rowspan = 2, colspan = 2) )
for a in ax:
    x,y = create_plots()
    a.plot(x,y)
plt.show()
```