In [5]:

```
# ASSIGNMENT
# input=s2b3c6
# output=ssbbbcccccc
a=input()
out=""
for i in a:
    if i.isalpha():
        alp=i
    else:
        dig=int(i)
        out=out+alp*(dig)
print(out)
```

```
ß2g4f6
ßßggggffffff
```

In [6]:

```
import numpy as np
k=np.arange(10)
print(k)
```

```
[0 1 2 3 4 5 6 7 8 9]
```

In [7]:

```
k=np.arange(10)
a=k.reshape(5,2)
b=k.dtype
print(b)
print(a)
```

```
int64
[[0 1]
 [2 3]
 [4 5]
 [6 7]
 [8 9]]
```

In [8]:

```
k=np.arange(10)
print(k.ndim)
```

```
1
```

In [9]:

```
kk=np.full((5,2),4)
print(kk)
```

```
[[4 4]
 [4 4]
 [4 4]
 [4 4]
 [4 4]]
```

In [10]:

```
rd=np.random.rand(5)
print(rd)
# we will get 5 no,s randomly
# only +ve no's
```

```
[0.36587369 0.36996983 0.03780866 0.26221188 0.40417844]
```

In [11]:

```
rd=np.random.randn(5)
print(rd)
# combination of +ve & -ve
```

```
[-0.85221942 -0.02291952 -0.06825319  0.82224313  0.62947253]
```

In [12]:

```
ktr=np.arange(12).reshape(3,4)
print(ktr)
```

```
[[ 0  1  2  3]
 [ 4  5  6  7]
 [ 8  9 10 11]]
```

In [13]:

```
ktr=np.arange(12).reshape(6,2)
print(ktr)
```

```
[[ 0  1]
 [ 2  3]
 [ 4  5]
 [ 6  7]
 [ 8  9]
 [10 11]]
```

In [14]:

```
ktr=np.arange(12).reshape(3,4)
print(ktr.min())
print(ktr.max())
print(ktr.std())
```

```
0
11
3.452052529534663
```

In [15]:

```
ktr=np.arange(12).reshape(6,2)
a=np.max(ktr,axis=1)
print(a)
```

```
[ 1  3  5  7  9 11]
```

In [16]:

```
ktr=np.arange(12).reshape(6,2)
a=np.min(ktr,axis=0)
print(a)
```

```
[0 1]
```

In [17]:

```
ln=np.linspace(1,100,5)
print(ln)
```

```
[  1.    25.75  50.5   75.25 100.  ]
```

PANDAS

- **pandas as data sets**
- **series--->1D**
- **dataframes--->2D**
- **datasets**

In [18]:

```
# pandas is lib from
import pandas as pd
print(dir(pd))
```

```
['BooleanDtype', 'Categorical', 'CategoricalDtype', 'CategoricalIndex', 'DataFrame', 'Dat
eOffset', 'DatetimeIndex', 'DatetimeTZDtype', 'ExcelFile', 'ExcelWriter', 'Float64Index',
'Grouper', 'HDFStore', 'Index', 'IndexSlice', 'Int16Dtype', 'Int32Dtype', 'Int64Dtype', '
Int64Index', 'Int8Dtype', 'Interval', 'IntervalDtype', 'IntervalIndex', 'MultiIndex', 'NA
', 'NaT', 'NamedAgg', 'Period', 'PeriodDtype', 'PeriodIndex', 'RangeIndex', 'Series', 'Sp
arseDtype', 'StringDtype', 'Timedelta', 'TimedeltaIndex', 'Timestamp', 'UInt16Dtype', 'UI
nt32Dtype', 'UInt64Dtype', 'UInt64Index', 'UInt8Dtype', '__builtins__', '__cached__', '__
doc__', '__docformat__', '__file__', '__getattr__', '__git_version__', '__loader__', '__n
ame__', '__package__', '__path__', '__spec__', '__version__', '_config', '_hashtable', '_
is_numpy_dev', '_lib', '_libs', '_np_version_under1p14', '_np_version_under1p15', '_np_ve
rsion_under1p16', '_np_version_under1p17', '_np_version_under1p18', '_testing', '_tslib',
'_typing', '_version', 'api', 'array', 'arrays', 'bdate_range', 'compat', 'concat', 'core
', 'crosstab', 'cut', 'date_range', 'describe_option', 'errors', 'eval', 'factorize', 'ge
t_dummies', 'get_option', 'infer_freq', 'interval_range', 'io', 'isna', 'isnull', 'json_n
ormalize', 'lreshape', 'melt', 'merge', 'merge_asof', 'merge_ordered', 'notna', 'notnull'
, 'offsets', 'option_context', 'options', 'pandas', 'period_range', 'pivot', 'pivot_table
', 'plotting', 'qcut', 'read_clipboard', 'read_csv', 'read_excel', 'read_feather', 'read_
fwf', 'read_gbq', 'read_hdf', 'read_html', 'read_json', 'read_orc', 'read_parquet', 'read
_pickle', 'read_sas', 'read_spss', 'read_sql', 'read_sql_query', 'read_sql_table', 'read_
stata', 'read_table', 'reset_option', 'set_eng_float_format', 'set_option', 'show_version
s', 'test', 'testing', 'timedelta_range', 'to_datetime', 'to_numeric', 'to_pickle', 'to_t
imedelta', 'tseries', 'unique', 'util', 'value_counts', 'wide_to_long']
```

In [19]:

```
a=pd.Series([1,2,3,4,5])# S is capital
print(a)
```

```
0    1
1    2
2    3
3    4
4    5
dtype: int64
```

In [20]:

```
slt=np.arange(5,10,1)
b=pd.Series(slt)
print(b)
print(b.values)
print(b.index)
```

```
0    5
1    6
2    7
3    8
4    9
dtype: int64
[5 6 7 8 9]
RangeIndex(start=0, stop=5, step=1)
```

In [21]:

```
a=pd.Series([1,2,3,4,5])
print(a.min())
print(a.max())
```

```
1
5
```

In [22]:

```
slt=np.arange(5,10,1)
c=pd.Series(slt,index=["a","b","c","d","e"])
print(c)
```

```
a    5
b    6
c    7
d    8
e    9
dtype: int64
```

In [23]:

```
print(c[4]==c["e"])
```

```
True
```

In [24]:

```
print(c[0]==c["a"])
```

```
True
```

In [25]:

```
print(c[c>6])
print(c[c>7])
print(c[4])
```

```
c    7
d    8
e    9
dtype: int64
d    8
e    9
dtype: int64
9
```

- **data frames--->2d array in pandas**
- **tabular data rows 7 coloumns**

In [26]:

```
ria={"name":["harsha","joel","mani","raja","vijay"],
     "dob":["2003","2002","2001","2004","2000"],
     "branch":["cse","it","ai","ds","ece"]
    }
df=pd.DataFrame(ria)
(df)
```

Out[26]:

	name	dob	branch
0	harsha	2003	cse
1	joel	2002	it
2	mani	2001	ai
3	raja	2004	ds
4	vijay	2000	ece

In [27]:

```
ria={"name":["harsha","joel","mani","raja","vijay"],
     "dob":["2003","2002","2001","2004","2000"],
     "branch":["cse","it","ai","ds","ece"]
    }
df=pd.DataFrame(ria)
(df.head(3))
```

Out[27]:

	name	dob	branch
0	harsha	2003	cse
1	joel	2002	it
2	mani	2001	ai

In [28]:

```
ria={"name":["harsha","joel","mani","raja","vijay"],
     "dob":["2003","2002","2001","2004","2000"],
     "branch":["cse","it","ai","ds","ece"]
    }
df=pd.DataFrame(ria)
(df.tail(3))
```

Out[28]:

	name	dob	branch
2	mani	2001	ai
3	raja	2004	ds
4	vijay	2000	ece

In [29]:

```
ria={"name":["harsha","joel","mani","raja","vijay"],
     "dob":["2003","2002","2001","2004","2000"],
     "branch":["cse","it","ai","ds","ece"]
    }
df=pd.DataFrame(ria)
df1=pd.DataFrame(df,columns=["name","dob","branch","marks"])
(df1)
```

Out[29]:

	name	dob	branch	marks
0	harsha	2003	cse	NaN
1	joel	2002	it	NaN
2	mani	2001	ai	NaN
3	raja	2004	ds	NaN
4	vijay	2000	ece	NaN

In [30]:

```
df1.isnull()
```

Out[30]:

	name	dob	branch	marks
0	False	False	False	True
1	False	False	False	True
2	False	False	False	True
3	False	False	False	True
4	False	False	False	True

In [31]:

```
df1["name"]
```

Out[31]:

```
0    harsha
1      joel
2      mani
3      raja
4     vijay
Name: name, dtype: object
```

In [32]:

```
df1["dob"]
```

Out[32]:

```
0    2003
1    2002
2    2001
3    2004
4    2000
Name: dob, dtype: object
```

In [33]:

```
df1["marks"]=[12,13,14,15,16]#np.random.rand(5)
df1
```

Out[33]:

	name	dob	branch	marks
0	harsha	2003	cse	12
1	joel	2002	it	13
2	mani	2001	ai	14
3	raja	2004	ds	15
4	vijay	2000	ece	16

In [34]:

```
df1.drop(4)
```

Out[34]:

	name	dob	branch	marks
0	harsha	2003	cse	12
1	joel	2002	it	13
2	mani	2001	ai	14
3	raja	2004	ds	15

In [35]:

```
ls#ksrcs.csv to check
```

```
Alarms/
Android/
Audiobooks/
ColorOS/
DCIM/
Data-Regstration.pdf
Documents/
Download/
Fonts/
JioSwitch/
Movies/
Music/
Notifications/
PDF'S/
PicsArt/
Pictures/
```

```
Pictures/
Podcasts/
Python Programming's Meeting Attendees.xls
Subtitles/
Untitled.ipynb
VidMate/
__pycache__/
apssdc/
iLovePDF/
inShare/
ksrcs.csv
ksrdata.pdf
oplus_log/
python/
```

In [36]:

```
#read_csv()
#read_json()
#read_excel()
```

In [38]:

```
data=pd.read_csv("ksrcs.csv")
data
```

Out[38]:

	Roll-No	Name	Python	C	Java	DS	OS	DBMS	Marks	Grade
0	20JR1A0590	M.Sushma	4	97	4	88	41	52	286	Pass
1	20JR1A0591	Nagumothu Navya	80	77	49	45	70	0	321	Pass
2	20JR1A0592	N.Lakshmi Vasavi	43	69	21	8	93	91	325	Pass
3	20JR1A0593	Sushma Namburi	34	72	97	42	33	71	349	Pass
4	20JR1A0594	N.lakshmi Chandana	93	95	14	10	73	20	305	Pass
...	...	...	...	...	...	...	...	...	...	...
156	20JR1A4442	VIVEK	66	87	2	79	92	16	342	Pass
157	20JR1AO5J2	Charan Siva Sai	73	58	33	56	56	19	295	Pass
158	20jr1A0518	anusha	33	61	26	53	29	18	220	Pass
159	20jr1a0552	Abc	68	26	18	20	24	70	226	Pass
160	20jr1ao5d8	hello	16	94	98	9	58	10	285	Pass

161 rows × 10 columns

In [39]:

```
data.head(10)
```

Out[39]:

	Roll-No	Name	Python	C	Java	DS	OS	DBMS	Marks	Grade
0	20JR1A0590	M.Sushma	4	97	4	88	41	52	286	Pass
1	20JR1A0591	Nagumothu Navya	80	77	49	45	70	0	321	Pass
2	20JR1A0592	N.Lakshmi Vasavi	43	69	21	8	93	91	325	Pass
3	20JR1A0593	Sushma Namburi	34	72	97	42	33	71	349	Pass
4	20JR1A0594	N.lakshmi Chandana	93	95	14	10	73	20	305	Pass
5	20JR1A0595	swapna	70	22	37	41	1	16	187	Fail
6	20JR1A0596	GSP SRIKANTH	95	65	2	86	91	66	405	Pass
7	20JR1A0597	prakash	72	42	54	45	98	84	395	Pass
8	20JR1A0598	harsha Ravuri	86	23	82	27	77	4	299	Pass

	Roll-No	Name	Python	C	Java	DS	OS	DBMS	Marks	Grade
9	20JR1A05A2	K Nagendra Babu	25	100	72	56	6	63	322	Pass

In [40]:

```
data.head(5)
```

Out[40]:

	Roll-No	Name	Python	C	Java	DS	OS	DBMS	Marks	Grade
0	20JR1A0590	M.Sushma	4	97	4	88	41	52	286	Pass
1	20JR1A0591	Nagumothu Navya	80	77	49	45	70	0	321	Pass
2	20JR1A0592	N.Lakshmi Vasavi	43	69	21	8	93	91	325	Pass
3	20JR1A0593	Sushma Namburi	34	72	97	42	33	71	349	Pass
4	20JR1A0594	N.lakshmi Chandana	93	95	14	10	73	20	305	Pass

In [41]:

```
data.tail(10)
```

Out[41]:

	Roll-No	Name	Python	C	Java	DS	OS	DBMS	Marks	Grade
151	20JR1A4362	SK.B.SHOHEB AKTHAR	81	95	27	27	34	88	352	Pass
152	20JR1A4363	MUNEER	46	57	55	9	70	50	287	Pass
153	20JR1A4364	Naga Sai Durgesh Singamsetty	65	62	9	54	46	67	303	Pass
154	20JR1A4431	Sakhamuri. Lathasri	65	89	57	32	82	80	405	Pass
155	20JR1A4435	Rushitha	17	61	83	12	3	67	243	Pass
156	20JR1A4442	VIVEK	66	87	2	79	92	16	342	Pass
157	20JR1AO5J2	Charan Siva Sai	73	58	33	56	56	19	295	Pass
158	20jr1A0518	anusha	33	61	26	53	29	18	220	Pass
159	20jr1a0552	Abc	68	26	18	20	24	70	226	Pass
160	20jr1ao5d8	hello	16	94	98	9	58	10	285	Pass

In [42]:

```
data.tail(5)
```

Out[42]:

	Roll-No	Name	Python	C	Java	DS	OS	DBMS	Marks	Grade
156	20JR1A4442	VIVEK	66	87	2	79	92	16	342	Pass
157	20JR1AO5J2	Charan Siva Sai	73	58	33	56	56	19	295	Pass
158	20jr1A0518	anusha	33	61	26	53	29	18	220	Pass
159	20jr1a0552	Abc	68	26	18	20	24	70	226	Pass
160	20jr1ao5d8	hello	16	94	98	9	58	10	285	Pass

In [43]:

```
data.min()
```

Out[43]:

```
Roll-No     20JR1A0590
Python               0
C                    0
Java                 0
DS                   1
```

```
OS                    0
DBMS                  0
Marks               107
Grade              Fail
dtype: object
```

In [44]:

```
data.max()
```

Out[44]:

```
Roll-No    20jr1ao5d8
Python             99
C                 100
Java              100
DS                100
OS                100
DBMS              100
Marks             441
Grade            Pass
dtype: object
```

In [45]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 161 entries, 0 to 160
Data columns (total 10 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   Roll-No  161 non-null    object
 1   Name     154 non-null    object
 2   Python   161 non-null    int64 
 3   C        161 non-null    int64 
 4   Java     161 non-null    int64 
 5   DS       161 non-null    int64 
 6   OS       161 non-null    int64 
 7   DBMS     161 non-null    int64 
 8   Marks    161 non-null    int64 
 9   Grade    161 non-null    object
dtypes: int64(7), object(3)
memory usage: 12.7+ KB
```

In [46]:

```
data.describe()
```

Out[46]:

	Python	C	Java	DS	OS	DBMS	Marks
count	161.000000	161.000000	161.000000	161.000000	161.000000	161.000000	161.000000
mean	51.664596	48.503106	45.776398	42.273292	52.093168	49.043478	289.354037
std	29.337677	29.097277	30.168894	28.250882	29.633132	29.139181	67.901161
min	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	107.000000
25%	27.000000	23.000000	20.000000	17.000000	29.000000	26.000000	234.000000
50%	50.000000	47.000000	49.000000	40.000000	52.000000	49.000000	295.000000
75%	80.000000	70.000000	69.000000	65.000000	79.000000	75.000000	331.000000
max	99.000000	100.000000	100.000000	100.000000	100.000000	100.000000	441.000000

In [47]:

```
cp=data["C"]# in file C is cap
cp
```

Out[47]:

```
0       97
1       77
2       69
3       72
4       95
        ..
156     87
157     58
158     61
159     26
160     94
Name: C, Length: 161, dtype: int64
```

In [48]:

```
ds=data["DS"]
ds
```

Out[48]:

```
0       88
1       45
2        8
3       42
4       10
        ..
156     79
157     56
158     53
159     20
160      9
Name: DS, Length: 161, dtype: int64
```

In [49]:

```
ja=data["Java"]
ja
```

Out[49]:

```
0        4
1       49
2       21
3       97
4       14
        ..
156      2
157     33
158     26
159     18
160     98
Name: Java, Length: 161, dtype: int64
```

In [50]:

```
py=data["Python"]
py
```

Out[50]:

```
0        4
1       80
2       43
3       34
4       93
        ..
156     66
157     73
158     33
159     68
160     16
Name: Python, Length: 161, dtype: int64
```

In [51]:

```
db=data["DBMS"]
db
```

Out[51]:

```
0      52
1       0
2      91
3      71
4      20
       ..
156    16
157    19
158    18
159    70
160    10
Name: DBMS, Length: 161, dtype: int64
```

In [52]:

```
os=data["OS"]
os
```

Out[52]:

```
0      41
1      70
2      93
3      33
4      73
       ..
156    92
157    56
158    29
159    24
160    58
Name: OS, Length: 161, dtype: int64
```

In [53]:

```
kk=data["DS"].drop(3)
print(kk)
```

```
0      88
1      45
2       8
4      10
5      41
       ..
156    79
157    56
158    53
159    20
160     9
Name: DS, Length: 160, dtype: int64
```

In [54]:

```
data.drop("Python",axis="columns",inplace=True)
print (data)
```

```
        Roll-No                Name   C  Java  DS  OS  DBMS  Marks Grade
0    20JR1A0590            M.Sushma  97     4  88  41    52    286  Pass
1    20JR1A0591     Nagumothu Navya  77    49  45  70     0    321  Pass
2    20JR1A0592    N.Lakshmi Vasavi  69    21   8  93    91    325  Pass
3    20JR1A0593      Sushma Namburi  72    97  42  33    71    349  Pass
4    20JR1A0594  N.lakshmi Chandana  95    14  10  73    20    305  Pass
..          ...                 ...  ..   ...  ..  ..   ...    ...   ...
156  20JR1A4442               VIVEK  87     2  79  92    16    342  Pass
```

```
157  20JR1AO5J2     Charan Siva Sai  58    33  56  56    19    295  Pass
158  20jr1A0518             anusha   61    26  53  29    18    220  Pass
159  20jr1a0552                Abc   26    18  20  24    70    226  Pass
160  20jr1ao5d8              hello   94    98   9  58    10    285  Pass

[161 rows x 9 columns]
```

In [55]:

```
data.drop("C",axis=1,inplace=True)
data
```

Out[55]:

	Roll-No	Name	Java	DS	OS	DBMS	Marks	Grade
0	20JR1A0590	M.Sushma	4	88	41	52	286	Pass
1	20JR1A0591	Nagumothu Navya	49	45	70	0	321	Pass
2	20JR1A0592	N.Lakshmi Vasavi	21	8	93	91	325	Pass
3	20JR1A0593	Sushma Namburi	97	42	33	71	349	Pass
4	20JR1A0594	N.lakshmi Chandana	14	10	73	20	305	Pass
...	...	...	...	...	...	...	...	...
156	20JR1A4442	VIVEK	2	79	92	16	342	Pass
157	20JR1AO5J2	Charan Siva Sai	33	56	56	19	295	Pass
158	20jr1A0518	anusha	26	53	29	18	220	Pass
159	20jr1a0552	Abc	18	20	24	70	226	Pass
160	20jr1ao5d8	hello	98	9	58	10	285	Pass

161 rows × 8 columns

In [10]:

```
import matplotlib.pyplot as plt
#pie
#bar
#list
#plot
#scattor
```

In [11]:

```
print(dir(plt))
```

```
['Annotation', 'Arrow', 'Artist', 'AutoLocator', 'Axes', 'Button', 'Circle', 'Figure', 'F
igureCanvasBase', 'FixedFormatter', 'FixedLocator', 'FormatStrFormatter', 'Formatter', 'F
uncFormatter', 'GridSpec', 'IndexLocator', 'Line2D', 'LinearLocator', 'Locator', 'LogForm
atter', 'LogFormatterExponent', 'LogFormatterMathtext', 'LogLocator', 'MaxNLocator', 'Mul
tipleLocator', 'Normalize', 'NullFormatter', 'NullLocator', 'Number', 'PolarAxes', 'Polyg
on', 'Rectangle', 'ScalarFormatter', 'Slider', 'Subplot', 'SubplotTool', 'Text', 'TickHel
per', 'Widget', '_INSTALL_FIG_OBSERVER', '_IP_REGISTERED', '__builtins__', '__cached__',
'__doc__', '__file__', '__loader__', '__name__', '__package__', '__spec__', '_auto_draw_i
f_interactive', '_backend_mod', '_get_running_interactive_framework', '_interactive_bk',
'_log', '_pylab_helpers', '_setp', '_setup_pyplot_info_docstrings', '_show', 'acorr', 'an
gle_spectrum', 'annotate', 'arrow', 'autoscale', 'autumn', 'axes', 'axhline', 'axhspan',
'axis', 'axvline', 'axvspan', 'bar', 'barbs', 'barh', 'bone', 'box', 'boxplot', 'broken_b
arh', 'cbook', 'cla', 'clabel', 'clf', 'clim', 'close', 'cm', 'cohere', 'colorbar', 'colo
rmaps', 'connect', 'contour', 'contourf', 'cool', 'copper', 'csd', 'cycler', 'dedent', 'd
elaxes', 'deprecated', 'disconnect', 'docstring', 'draw', 'draw_all', 'draw_if_interactiv
e', 'errorbar', 'eventplot', 'figaspect', 'figimage', 'figlegend', 'fignum_exists', 'figt
ext', 'figure', 'fill', 'fill_between', 'fill_betweenx', 'findobj', 'flag', 'functools',
'gca', 'gcf', 'gci', 'get', 'get_backend', 'get_cmap', 'get_current_fig_manager', 'get_fi
glabels', 'get_fignums', 'get_plot_commands', 'get_scale_docs', 'get_scale_names', 'getp'
, 'ginput', 'gray', 'grid', 'hexbin', 'hist', 'hist2d', 'hlines', 'hot', 'hsv', 'importli
b', 'imread', 'imsave', 'imshow', 'inferno', 'inspect', 'install_repl_displayhook', 'inte
ractive', 'ioff', 'ion', 'isinteractive', 'jet', 'legend', 'locator_params', 'logging', '
loglog', 'magma', 'magnitude_spectrum', 'margins', 'matplotlib', 'matshow', 'minorticks_o
```

```
logit', 'magma', 'magnitude_spectrum', 'margins', 'matplotlib', 'matshow', 'minorticks_o
ff', 'minorticks_on', 'mlab', 'new_figure_manager', 'nipy_spectral', 'np', 'pause', 'pcol
or', 'pcolormesh', 'phase_spectrum', 'pie', 'pink', 'plasma', 'plot', 'plot_date', 'plotf
ile', 'plotting', 'polar', 'prism', 'psd', 'quiver', 'quiverkey', 'rc', 'rcParams', 'rcPa
ramsDefault', 'rcParamsOrig', 'rc_context', 'rcdefaults', 'rcsetup', 're', 'register_cmap
', 'rgrids', 'savefig', 'sca', 'scatter', 'sci', 'semilogx', 'semilogy', 'set_cmap', 'set
_loglevel', 'setp', 'show', 'silent_list', 'specgram', 'spring', 'spy', 'stackplot', 'ste
m', 'step', 'streamplot', 'style', 'subplot', 'subplot2grid', 'subplot_tool', 'subplots',
'subplots_adjust', 'summer', 'suptitle', 'switch_backend', 'sys', 'table', 'text', 'theta
grids', 'tick_params', 'ticklabel_format', 'tight_layout', 'time', 'title', 'tricontour',
'tricontourf', 'tripcolor', 'triplot', 'twinx', 'twiny', 'uninstall_repl_displayhook', 'v
iolinplot', 'viridis', 'vlines', 'waitforbuttonpress', 'warn_deprecated', 'winter', 'xcor
r', 'xkcd', 'xlabel', 'xlim', 'xscale', 'xticks', 'ylabel', 'ylim', 'yscale', 'yticks']
```

In [6]:

```
import numpy as np
x=np.arange(10)
y=np.arange(20,40,2)
```

In [12]:

```
plt.plot(x,y)
plt.show()
```