Agenda

- **Universal Functions (ufunc) on 2D**✅
  - `np.unique()` ✅
  - `np.where()` ✅
- **Use Case: Fitness Data analysis**✅
  - Loading data set and EDA using numpy
  - `np.argmax()` ✅
- **Vectorization**✅
  - `np.vectorize()` ✅
- **3D arrays**✅
- **Use Case: Image Manipulation using Numpy**
  - Opening an Image
  - Details of an image
  - Visualizing Channels
  - Rotating an Image (Transposing a Numpy Array)
  - Trim image
- **Broadcasting**
- **Array splitting and Merging**
  - Splitting arrays - `split()`, `hsplit()`, `vsplit()`
  - Merging Arrays - `hstack()`, `vstack()`

Universal Functions (ufunc) on 2D

`np.unique()`

In [1]:
```
data=np.loadtxt("/Users/nikhilsanghi/Downloads/01_dsml-course-main-live/batches/2_Sept_Beg_Tue_Oct_Beg_Tue/04_Numpy_4/survey.txt")
data
```
Out[1]:
```
array([ 7., 10.,  5., ...,  5.,  9., 10.])
```

In [2]:
```
np.unique(data)
```
Out[2]:
```
array([ 1.,  4.,  5.,  7.,  8.,  9., 10.])
```

In [3]:
```
np.unique(data,return_counts=True)
```
Out[3]:
```
(array([ 1.,  4.,  5.,  7.,  8.,  9., 10.]),
 array([117, 112, 103, 123, 103, 394, 215]))
```

`np.where()`

In [4]:
```
a=np.array([ 2,  1,  6,  4,  7,  3,  8,  9, 11])
a
```
Out[4]:
```
array([ 2,  1,  6,  4,  7,  3,  8,  9, 11])
```

In [5]:
```
a<6
```
Out[5]:
```
array([ True,  True, False,  True, False,  True, False, False, False])
```

In [6]:
```
a[a<6]
```
Out[6]:
```
array([2, 1, 4, 3])
```

In [7]:
```
np.where(a<6)
```
Out[7]:
```
(array([0, 1, 3, 5]),)
```

In [8]:
```
np.where(a<6,-1,+1)
```
Out[8]:
```
array([-1, -1,  1, -1,  1, -1,  1,  1,  1])
```

In [9]:
```
np.where(a<6,-1,a)
```
Out[9]:
```
array([-1, -1,  6, -1,  7, -1,  8,  9, 11])
```

Use Case: Fitness Data analysis

In [89]:
```
data=np.loadtxt("/Users/nikhilsanghi/Downloads/01_dsml-course-main-live/batches/2_Sept_Beg_Tue_Oct_Beg_Tue/04_Numpy_4/fit.txt",dtype="str")
data
```
Out[89]:
```
array([['06-10-2017', '5464', 'Neutral', '181', '5', 'Inactive'],
       ['07-10-2017', '6041', 'Sad', '197', '8', 'Inactive'],
       ['08-10-2017', '25', 'Sad', '0', '5', 'Inactive'],
       ['09-10-2017', '5461', 'Sad', '174', '4', 'Inactive'],
       ['10-10-2017', '6915', 'Neutral', '223', '5', 'Active'],
       ['11-10-2017', '4545', 'Sad', '149', '6', 'Inactive'],
       ['12-10-2017', '4340', 'Sad', '140', '6', 'Inactive'],
       ['13-10-2017', '1230', 'Sad', '38', '7', 'Inactive'],
       ['14-10-2017', '61', 'Sad', '1', '5', 'Inactive'],
       ['15-10-2017', '1258', 'Sad', '40', '6', 'Inactive'],
       ['16-10-2017', '3148', 'Sad', '101', '8', 'Inactive'],
       ['17-10-2017', '4687', 'Sad', '152', '5', 'Inactive'],
       ['18-10-2017', '4732', 'Happy', '150', '6', 'Active'],
       ['19-10-2017', '3519', 'Sad', '113', '7', 'Inactive'],
       ['20-10-2017', '1580', 'Sad', '49', '5', 'Inactive'],
       ['21-10-2017', '2822', 'Sad', '86', '6', 'Inactive'],
       ['22-10-2017', '181', 'Sad', '6', '8', 'Inactive'],
       ['23-10-2017', '3158', 'Neutral', '99', '5', 'Inactive'],
       ['24-10-2017', '4383', 'Neutral', '143', '4', 'Inactive'],
       ['25-10-2017', '3881', 'Neutral', '125', '5', 'Inactive'],
       ['26-10-2017', '4037', 'Neutral', '129', '6', 'Inactive'],
       ['27-10-2017', '202', 'Neutral', '6', '8', 'Inactive'],
       ['28-10-2017', '292', 'Neutral', '9', '5', 'Inactive'],
       ['29-10-2017', '330', 'Happy', '10', '6', 'Inactive'],
       ['30-10-2017', '2209', 'Neutral', '72', '5', 'Inactive'],
       ['31-10-2017', '4550', 'Happy', '150', '8', 'Active'],
       ['01-11-2017', '4435', 'Happy', '141', '5', 'Inactive'],
       ['02-11-2017', '4779', 'Happy', '156', '4', 'Inactive'],
       ['03-11-2017', '1831', 'Happy', '57', '5', 'Inactive'],
       ['04-11-2017', '2255', 'Happy', '72', '4', 'Inactive'],
       ['05-11-2017', '539', 'Happy', '17', '5', 'Active'],
       ['06-11-2017', '5464', 'Happy', '181', '4', 'Inactive'],
       ['07-11-2017', '6041', 'Neutral', '197', '3', 'Inactive'],
       ['08-11-2017', '4068', 'Happy', '131', '2', 'Inactive'],
       ['09-11-2017', '4683', 'Happy', '154', '9', 'Inactive'],
       ['10-11-2017', '4033', 'Happy', '137', '5', 'Inactive'],
       ['11-11-2017', '6314', 'Happy', '193', '6', 'Active'],
       ['12-11-2017', '614', 'Happy', '19', '4', 'Active'],
       ['13-11-2017', '3149', 'Happy', '101', '5', 'Active'],
       ['14-11-2017', '4005', 'Happy', '139', '8', 'Active'],
       ['15-11-2017', '4880', 'Happy', '164', '4', 'Active'],
       ['16-11-2017', '4136', 'Happy', '137', '5', 'Active'],
       ['17-11-2017', '705', 'Happy', '22', '6', 'Active'],
       ['18-11-2017', '570', 'Neutral', '17', '5', 'Active'],
       ['19-11-2017', '269', 'Happy', '9', '6', 'Active'],
       ['20-11-2017', '4275', 'Happy', '145', '5', 'Inactive'],
       ['21-11-2017', '5999', 'Happy', '192', '6', 'Inactive'],
       ['22-11-2017', '4421', 'Happy', '146', '5', 'Inactive'],
       ['23-11-2017', '6930', 'Happy', '234', '6', 'Inactive'],
       ['24-11-2017', '5195', 'Happy', '167', '5', 'Inactive'],
       ['25-11-2017', '546', 'Happy', '16', '6', 'Inactive'],
       ['26-11-2017', '493', 'Happy', '17', '7', 'Active'],
       ['27-11-2017', '995', 'Happy', '32', '6', 'Active'],
       ['28-11-2017', '1163', 'Neutral', '35', '7', 'Active'],
       ['29-11-2017', '6676', 'Sad', '220', '6', 'Active'],
       ['30-11-2017', '3608', 'Happy', '116', '5', 'Active'],
       ['01-12-2017', '774', 'Happy', '23', '6', 'Active'],
       ['02-12-2017', '1421', 'Happy', '44', '7', 'Active'],
       ['03-12-2017', '4064', 'Happy', '131', '8', 'Active'],
       ['04-12-2017', '2725', 'Happy', '86', '8', 'Active'],
       ['05-12-2017', '5934', 'Happy', '194', '7', 'Active'],
       ['06-12-2017', '1867', 'Happy', '60', '8', 'Active'],
       ['07-12-2017', '3721', 'Sad', '121', '5', 'Active'],
       ['08-12-2017', '2374', 'Neutral', '76', '4', 'Inactive'],
       ['09-12-2017', '2909', 'Neutral', '93', '3', 'Active'],
       ['10-12-2017', '1648', 'Sad', '53', '3', 'Active'],
       ['11-12-2017', '799', 'Sad', '25', '4', 'Inactive'],
       ['12-12-2017', '7102', 'Neutral', '227', '5', 'Active'],
       ['13-12-2017', '3941', 'Neutral', '125', '5', 'Active'],
       ['14-12-2017', '7422', 'Happy', '243', '5', 'Active'],
       ['15-12-2017', '437', 'Neutral', '14', '3', 'Active'],
       ['16-12-2017', '1231', 'Neutral', '39', '4', 'Active'],
       ['17-12-2017', '1696', 'Sad', '55', '4', 'Inactive'],
       ['18-12-2017', '4921', 'Neutral', '158', '5', 'Active'],
       ['19-12-2017', '221', 'Sad', '7', '5', 'Active'],
       ['20-12-2017', '6500', 'Neutral', '213', '5', 'Active'],
       ['21-12-2017', '3575', 'Neutral', '116', '5', 'Active'],
       ['22-12-2017', '4061', 'Sad', '129', '5', 'Inactive'],
       ['23-12-2017', '651', 'Sad', '21', '5', 'Inactive'],
       ['24-12-2017', '753', 'Sad', '28', '4', 'Inactive'],
       ['25-12-2017', '518', 'Sad', '16', '3', 'Inactive'],
       ['26-12-2017', '5537', 'Happy', '180', '4', 'Active'],
       ['27-12-2017', '4108', 'Neutral', '138', '5', 'Active'],
       ['28-12-2017', '5376', 'Happy', '176', '5', 'Active'],
       ['29-12-2017', '3066', 'Neutral', '99', '4', 'Active'],
       ['30-12-2017', '177', 'Sad', '5', '5', 'Inactive'],
       ['31-12-2017', '36', 'Sad', '1', '3', 'Inactive'],
       ['01-01-2018', '299', 'Sad', '10', '3', 'Inactive'],
       ['02-01-2018', '1447', 'Neutral', '47', '3', 'Inactive'],
       ['03-01-2018', '2599', 'Neutral', '84', '2', 'Inactive'],
       ['04-01-2018', '702', 'Sad', '23', '3', 'Inactive'],
       ['05-01-2018', '133', 'Sad', '4', '2', 'Inactive'],
       ['06-01-2018', '153', 'Happy', '0', '8', 'Inactive'],
       ['07-01-2018', '500', 'Neutral', '0', '5', 'Active'],
       ['08-01-2018', '2127', 'Neutral', '0', '5', 'Inactive'],
       ['09-01-2018', '2203', 'Happy', '0', '5', 'Active']], dtype='<U10')
```

In [11]:
```
date=data[::,0]
step_counts=data[::,1]
mood=data[::,2]
calories_burnt=data[::,3]
hours_of_sleep=data[::,4]
activity=data[::,5]
```

In [12]:
```
date
```
Out[12]:
```
array(['06-10-2017', '07-10-2017', '08-10-2017', '09-10-2017',
       '10-10-2017', '11-10-2017', '12-10-2017', '13-10-2017',
       '14-10-2017', '15-10-2017', '16-10-2017', '17-10-2017',
       '18-10-2017', '19-10-2017', '20-10-2017', '21-10-2017',
       '22-10-2017', '23-10-2017', '24-10-2017', '25-10-2017',
       '26-10-2017', '27-10-2017', '28-10-2017', '29-10-2017',
       '30-10-2017', '31-10-2017', '01-11-2017', '02-11-2017',
       '03-11-2017', '04-11-2017', '05-11-2017', '06-11-2017',
       '07-11-2017', '08-11-2017', '09-11-2017', '10-11-2017',
       '11-11-2017', '12-11-2017', '13-11-2017', '14-11-2017',
       '15-11-2017', '16-11-2017', '17-11-2017', '18-11-2017',
       '19-11-2017', '20-11-2017', '21-11-2017', '22-11-2017',
       '23-11-2017', '24-11-2017', '25-11-2017', '26-11-2017',
       '27-11-2017', '28-11-2017', '29-11-2017', '30-11-2017',
       '01-12-2017', '02-12-2017', '03-12-2017', '04-12-2017',
       '05-12-2017', '06-12-2017', '07-12-2017', '08-12-2017',
       '09-12-2017', '10-12-2017', '11-12-2017', '12-12-2017',
       '13-12-2017', '14-12-2017', '15-12-2017', '16-12-2017',
       '17-12-2017', '18-12-2017', '19-12-2017', '20-12-2017',
       '21-12-2017', '22-12-2017', '23-12-2017', '24-12-2017',
       '25-12-2017', '26-12-2017', '27-12-2017', '28-12-2017',
       '29-12-2017', '30-12-2017', '31-12-2017', '01-01-2018',
       '02-01-2018', '03-01-2018', '04-01-2018', '05-01-2018',
       '06-01-2018', '07-01-2018', '08-01-2018', '09-01-2018'],
      dtype='<U10')
```

In [13]:
```
step_counts
```
Out[13]:
```
array(['5464', '6041', '25', '5461', '6915', '4545', '4340', '1230', '61',
       '1258', '3148', '4687', '4732', '3519', '1580', '2822', '181',
       '3158', '4383', '3881', '4037', '202', '292', '330', '2209',
       '4550', '4435', '4779', '1831', '2255', '539', '5464', '6041',
       '4068', '4683', '4033', '6314', '614', '3149', '4005', '4880',
       '4136', '705', '570', '269', '4275', '5999', '4421', '6930',
       '5195', '546', '493', '995', '1163', '6676', '3608', '774', '1421',
       '4064', '2725', '5934', '1867', '3721', '2374', '2909', '1648',
       '799', '7102', '3941', '7422', '437', '1231', '1696', '4921',
       '221', '6500', '3575', '4061', '651', '753', '518', '5537', '4108',
       '5376', '3066', '177', '36', '299', '1447', '2599', '702', '133',
       '153', '500', '2127', '2203'], dtype='<U10')
```

In [16]:
```
step_counts=np.array(step_counts,dtype="int")
step_counts
```
Out[16]:
```
array([5464, 6041,   25, 5461, 6915, 4545, 4340, 1230,   61, 1258, 3148,
       4687, 4732, 3519, 1580, 2822,  181, 3158, 4383, 3881, 4037,  202,
        292,  330, 2209, 4550, 4435, 4779, 1831, 2255,  539, 5464, 6041,
       4068, 4683, 4033, 6314,  614, 3149, 4005, 4880, 4136,  705,  570,
        269, 4275, 5999, 4421, 6930, 5195,  546,  493,  995, 1163, 6676,
       3608,  774, 1421, 4064, 2725, 5934, 1867, 3721, 2374, 2909, 1648,
        799, 7102, 3941, 7422,  437, 1231, 1696, 4921,  221, 6500, 3575,
       4061,  651,  753,  518, 5537, 4108, 5376, 3066,  177,   36,  299,
       1447, 2599,  702,  133,  153,  500, 2127, 2203])
```

In [17]:
```
calories_burnt=np.array(calories_burnt,dtype="int")
calories_burnt
```
Out[17]:
```
array([181, 197,   0, 174, 223, 149, 140,  38,   1,  40, 101, 152, 150,
       113,  49,  86,   6,  99, 143, 125, 129,   6,   9,  10,  72, 150,
       141, 156,  57,  72,  17, 181, 197, 131, 154, 137, 193,  19, 101,
       139, 164, 137,  22,  17,   9, 145, 192, 146, 234, 167,  16,  17,
        32,  35, 220, 116,  23,  44, 131,  86, 194,  60, 121,  76,  93,
        53,  25, 227, 125, 243,  14,  39,  55, 158,   7, 213, 116, 129,
        21,  28,  16, 180, 138, 176,  99,   5,   1,  10,  47,  84,  23,
         4,   0,   0,   0,   0])
```

In [18]:
```
hours_of_sleep=np.array(hours_of_sleep,dtype="int")
hours_of_sleep
```
Out[18]:
```
array([5, 8, 5, 4, 5, 6, 6, 7, 5, 6, 8, 5, 6, 7, 5, 6, 8, 5, 4, 5, 6, 8,
       5, 6, 5, 8, 5, 4, 5, 4, 5, 4, 3, 2, 9, 5, 6, 4, 5, 8, 4, 5, 6, 5,
       6, 5, 6, 5, 6, 5, 6, 7, 6, 7, 6, 5, 6, 7, 8, 8, 7, 8, 5, 4, 3, 3,
       4, 5, 5, 5, 3, 4, 4, 5, 5, 5, 5, 5, 5, 4, 3, 4, 5, 5, 4, 5, 3, 3,
       3, 2, 3, 2, 8, 5, 5, 5])
```

In [19]:
```
mood
```
Out[19]:
```
array(['Neutral', 'Sad', 'Sad', 'Sad', 'Neutral', 'Sad', 'Sad', 'Sad',
       'Sad', 'Sad', 'Sad', 'Sad', 'Happy', 'Sad', 'Sad', 'Sad', 'Sad',
       'Neutral', 'Neutral', 'Neutral', 'Neutral', 'Neutral', 'Neutral',
       'Happy', 'Neutral', 'Happy', 'Happy', 'Happy', 'Happy', 'Happy',
       'Happy', 'Happy', 'Neutral', 'Happy', 'Happy', 'Happy', 'Happy',
       'Happy', 'Happy', 'Happy', 'Happy', 'Happy', 'Happy', 'Neutral',
       'Happy', 'Happy', 'Happy', 'Happy', 'Happy', 'Happy', 'Happy',
       'Happy', 'Happy', 'Neutral', 'Sad', 'Happy', 'Happy', 'Happy',
       'Happy', 'Happy', 'Happy', 'Happy', 'Sad', 'Neutral', 'Neutral',
       'Sad', 'Sad', 'Neutral', 'Neutral', 'Happy', 'Neutral', 'Neutral',
       'Sad', 'Neutral', 'Sad', 'Neutral', 'Neutral', 'Sad', 'Sad', 'Sad',
       'Sad', 'Happy', 'Neutral', 'Happy', 'Neutral', 'Sad', 'Sad', 'Sad',
       'Neutral', 'Neutral', 'Sad', 'Sad', 'Happy', 'Neutral', 'Neutral',
       'Happy'], dtype='<U10')
```

In [20]:
```
activity
```
Out[20]:
```
array(['Inactive', 'Inactive', 'Inactive', 'Inactive', 'Active',
       'Inactive', 'Inactive', 'Inactive', 'Inactive', 'Inactive',
       'Inactive', 'Inactive', 'Active', 'Inactive', 'Inactive',
       'Inactive', 'Inactive', 'Inactive', 'Inactive', 'Inactive',
       'Inactive', 'Inactive', 'Inactive', 'Inactive', 'Inactive',
       'Active', 'Inactive', 'Inactive', 'Inactive', 'Inactive', 'Active',
       'Inactive', 'Inactive', 'Inactive', 'Inactive', 'Inactive',
       'Active', 'Active', 'Active', 'Active', 'Active', 'Active',
       'Active', 'Active', 'Active', 'Inactive', 'Inactive', 'Inactive',
       'Inactive', 'Inactive', 'Inactive', 'Active', 'Active', 'Active',
       'Active', 'Active', 'Active', 'Active', 'Active', 'Active',
       'Active', 'Active', 'Active', 'Inactive', 'Active', 'Active',
       'Inactive', 'Active', 'Active', 'Active', 'Active', 'Active',
       'Inactive', 'Active', 'Active', 'Active', 'Active', 'Inactive',
       'Inactive', 'Inactive', 'Inactive', 'Active', 'Active', 'Active',
       'Active', 'Inactive', 'Inactive', 'Inactive', 'Inactive',
       'Inactive', 'Inactive', 'Inactive', 'Inactive', 'Active',
       'Inactive', 'Active'], dtype='<U10')
```

In [21]:
```
np.mean(step_counts)
```
Out[21]:
```
2935.9375
```

In [23]:
```
np.unique(mood,return_counts=True)
```
Out[23]:
```
(array(['Happy', 'Neutral', 'Sad'], dtype='<U10'), array([40, 27, 29]))
```

In [24]:
```
np.unique(activity,return_counts=True)
```
Out[24]:
```
(array(['Active', 'Inactive'], dtype='<U10'), array([42, 54]))
```

In [25]:
```
np.max(step_counts)
```
Out[25]:
```
7422
```

In [26]:
```
np.argmax(step_counts)
```
Out[26]:
```
69
```

In [27]:
```
date[69]
```
Out[27]:
```
'14-12-2017'
```

In [28]:
```
date[np.argmax(step_counts)]
```
Out[28]:
```
'14-12-2017'
```

In [29]:
```
mood=="Sad"
```
Out[29]:
```
array([False,  True,  True,  True, False,  True,  True,  True,  True,
        True,  True,  True, False,  True,  True,  True,  True, False,
       False, False, False, False, False, False, False, False, False,
       False, False, False, False, False, False, False, False, False,
       False, False, False, False, False, False, False, False, False,
       False, False, False, False, False, False, False, False, False,
        True, False, False, False, False, False, False, False,  True,
       False, False,  True,  True, False, False, False, False, False,
        True, False,  True, False, False,  True,  True,  True,  True,
       False, False, False, False,  True,  True,  True, False, False,
        True,  True, False, False, False, False])
```

In [30]:
```
step_counts[mood=="Sad"]
```
Out[30]:
```
array([6041,   25, 5461, 4545, 4340, 1230,   61, 1258, 3148, 4687, 3519,
       1580, 2822,  181, 6676, 3721, 1648,  799, 1696,  221, 4061,  651,
        753,  518,  177,   36,  299,  702,  133])
```

In [31]:
```
np.mean(step_counts[mood=="Sad"])
```
Out[31]:
```
2103.0689655172414
```

In [32]:
```
np.mean(step_counts[mood=="Neutral"])
```
Out[32]:
```
3153.777777777778
```

In [34]:
```
np.mean(step_counts[mood=="Happy"])
```
Out[34]:
```
3392.725
```

In [35]:
```
step_counts>4000
```
Out[35]:
```
array([ True,  True, False,  True,  True,  True,  True, False, False,
       False, False,  True,  True, False, False, False, False, False,
        True, False,  True, False, False, False, False,  True,  True,
        True, False, False, False,  True,  True,  True,  True,  True,
        True, False, False,  True,  True,  True, False, False, False,
        True,  True,  True,  True,  True, False, False, False, False,
        True, False, False, False,  True, False,  True, False, False,
       False, False, False, False,  True, False,  True, False, False,
       False,  True, False,  True, False,  True, False, False, False,
        True,  True,  True, False, False, False, False, False, False,
       False, False, False, False, False, False])
```

In [36]:
```
mood[step_counts>4000]
```
Out[36]:
```
array(['Neutral', 'Sad', 'Sad', 'Neutral', 'Sad', 'Sad', 'Sad', 'Happy',
       'Neutral', 'Neutral', 'Happy', 'Happy', 'Happy', 'Happy',
       'Neutral', 'Happy', 'Happy', 'Happy', 'Happy', 'Happy', 'Happy',
       'Happy', 'Happy', 'Happy', 'Happy', 'Happy', 'Happy', 'Sad',
       'Happy', 'Happy', 'Neutral', 'Happy', 'Neutral', 'Neutral', 'Sad',
       'Happy', 'Neutral', 'Happy'], dtype='<U10')
```

In [38]:
```
np.unique(mood[step_counts>4000],return_counts=True)
```
Out[38]:
```
(array(['Happy', 'Neutral', 'Sad'], dtype='<U10'), array([22,  9,  7]))
```

In [39]:
```
np.unique(mood[step_counts<2000],return_counts=True)
```
Out[39]:
```
(array(['Happy', 'Neutral', 'Sad'], dtype='<U10'), array([13,  8, 18]))
```

In [40]:
```
np.unique(mood[hours_of_sleep<4],return_counts=True)
```
Out[40]:
```
(array(['Happy', 'Neutral', 'Sad'], dtype='<U10'), array([1, 5, 6]))
```

In [42]:
```
np.unique(mood[hours_of_sleep>5],return_counts=True)
```
Out[42]:
```
(array(['Happy', 'Neutral', 'Sad'], dtype='<U10'), array([20,  3, 10]))
```

In [45]:
```
np.unique(mood[activity=="Inactive"],return_counts=True)
```
Out[45]:
```
(array(['Happy', 'Neutral', 'Sad'], dtype='<U10'), array([16, 13, 25]))
```

In [46]:
```
np.unique(mood[activity=="Active"],return_counts=True)
```
Out[46]:
```
(array(['Happy', 'Neutral', 'Sad'], dtype='<U10'), array([24, 14,  4]))
```

In [48]:
```
np.mean(hours_of_sleep[mood=="Happy"])
```
Out[48]:
```
5.725
```

In [49]:
```
np.mean(hours_of_sleep[mood=="Sad"])
```
Out[49]:
```
5.0344827586206895
```

In [ ]:

Vectorization

In [53]:
```
a=np.arange(1,11)
a
```
Out[53]:
```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

In [54]:
```
import math
math.log
```
Out[54]:
```
<function math.log>
```

In [55]:
```
l=[ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10]
l_e=[]
for i in l:
    l_e.append(math.log(i))
l_e
```
Out[55]:
```
[0.0,
 0.6931471805599453,
 1.0986122886681098,
 1.3862943611198906,
 1.6094379124341003,
 1.791759469228055,
 1.9459101490553132,
 2.0794415416798357,
 2.1972245773362196,
 2.302585092994046]
```

In [56]:
```
np.log(a)
```
Out[56]:
```
array([0.        , 0.69314718, 1.09861229, 1.38629436, 1.60943791,
       1.79175947, 1.94591015, 2.07944154, 2.19722458, 2.30258509])
```

In [58]:
```
y=np.vectorize(math.log)
y(a)
```
Out[58]:
```
array([0.        , 0.69314718, 1.09861229, 1.38629436, 1.60943791,
       1.79175947, 1.94591015, 2.07944154, 2.19722458, 2.30258509])
```

In [59]:
```
math.log(a)
```
```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/var/folders/hd/9z4dczb56dj54lb7q8w7s4zw0000gn/T/ipykernel_40565/1428990103.py in <module>
----> 1 math.log(a)

TypeError: only size-1 arrays can be converted to Python scalars
```

In [66]:
```
def custom_function(x):
    if(i%2==0):
        return x**2
    else:
        return x**3
```

In [60]:
```
a
```
Out[60]:
```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

In [63]:
```
l=[ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10]
l_e=[]
for i in l:
    l_e.append(custom_function(i))
l_e
```
Out[63]:
```
[1, 4, 27, 16, 125, 36, 343, 64, 729, 100]
```

In [72]:
```
a
```
Out[72]:
```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10])
```

In [69]:
```
def custom_function(x):
    if(x%2==0):
        return x**2
    else:
        return x**3
```

In [70]:
```
y=np.vectorize(custom_function)
y
```
Out[70]:
```
<numpy.vectorize at 0x7fdb40ae62b0>
```

In [71]:
```
y(a)
```
Out[71]:
```
array([  1,   4,  27,  16, 125,  36, 343,  64, 729, 100])
```

In [ ]:

3D arrays

In [ ]:
```
# 1 Dimensional Arrays --> Vectors
# 2 Dimensional Arrays --> Matrix
# 3 Dimensional Arrays --> Tensors
```

In [73]:
```
a=np.arange(1,25).reshape((2,3,4))
a
```
Out[73]:
```
array([[[ 1,  2,  3,  4],
        [ 5,  6,  7,  8],
        [ 9, 10, 11, 12]],

       [[13, 14, 15, 16],
        [17, 18, 19, 20],
        [21, 22, 23, 24]]])
```

In [74]:
```
a.ndim
```
Out[74]:
```
3
```

In [75]:
```
a.shape
```
Out[75]:
```
(2, 3, 4)
```

In [76]:
```
a.size
```
Out[76]:
```
24
```

In [77]:
```
len(a)
```
Out[77]:
```
2
```

In [78]:
```
a=np.array([[[ 100,  15, 25,  36],[ 1,  51, 62,  77],[ 81,  9,101, 11]],[[12, 130,144, 155],[196, 17,8, 19],[2, 21,225, 23]]])
a
```
Out[78]:
```
array([[[100,  15,  25,  36],
        [  1,  51,  62,  77],
        [ 81,   9, 101,  11]],

       [[ 12, 130, 144, 155],
        [196,  17,   8,  19],
        [  2,  21, 225,  23]]])
```

In [79]:
```
np.max(a,axis=0)
```
Out[79]:
```
array([[100, 130, 144, 155],
       [196,  51,  62,  77],
       [ 81,  21, 225,  23]])
```

In [80]:
```
np.max(a,axis=-3)
```
Out[80]:
```
array([[100, 130, 144, 155],
       [196,  51,  62,  77],
       [ 81,  21, 225,  23]])
```

In [82]:
```
np.min(a,axis=1)
```
Out[82]:
```
array([[ 1,  9, 25, 11],
       [ 2, 17,  8, 19]])
```

In [83]:
```
np.min(a,axis=-2)
```
Out[83]:
```
array([[ 1,  9, 25, 11],
       [ 2, 17,  8, 19]])
```

In [85]:
```
np.sort(a,axis=2)
```
Out[85]:
```
array([[[ 15,  25,  36, 100],
        [  1,  51,  62,  77],
        [  9,  11,  81, 101]],

       [[ 12, 130, 144, 155],
        [  8,  17,  19, 196],
        [  2,  21,  23, 225]]])
```

In [86]:
```
np.sort(a,axis=-1)
```
Out[86]:
```
array([[[ 15,  25,  36, 100],
        [  1,  51,  62,  77],
        [  9,  11,  81, 101]],

       [[ 12, 130, 144, 155],
        [  8,  17,  19, 196],
        [  2,  21,  23, 225]]])
```

Use Case: Image Manipulation using Numpy

In [ ]:

In [ ]: