In [1]:
```
from math import sin, cos, sqrt, atan2, radians, degrees
```

1 Calculate the Radius of the Earth for a given Latitude

In [2]:
```
def earthRadius(latitude, elevation = 0):
    '''Provides Earth's Radius (in miles) at the provided latitude (in degrees).
Includes option to include an elevation (in feet).'''
    lat = radians(latitude)
    elev = elevation/5280    # elevation in miles from feet
    a = 3963.191    # Earth's radius at equator (miles)
    b = 3949.903    # Earth's radius at poles (miles)

    Rsl = sqrt( ((a**2 * cos(lat))**2 + (b**2 * sin(lat))**2) / ((a * cos(lat))**2 + (
    # ^ calculates radius at sea level

    R = Rsl + elev

    return R
```

In [3]:
```
earthRadius(29.3, 200)
```

Out[3]: 3960.0667413903634

2 Calculate Distance (Lat-Long)

- Requires two coordinates (Lat, Long) in degree format
- Requires `earthRadius()` function as a prerequisite
- Uses sea level (elevation = 0) by default unless specified otherwise (usually a minor impact on outcome)

In [4]:
```
def distanceLatLong(Lat1, Long1, Lat2, Long2, elev1 = 0, elev2 = 0):
    '''Provides distance (in feet) between two points (coordinates in degrees of latit
    R1 = earthRadius(Lat1,elev1)
    R2 = earthRadius(Lat2,elev2)
    R = (R1 + R2)/2    # average Radius of Earth

    lat1 = radians(Lat1)    # convert to radians
    lon1 = radians(Long1)
    lat2 = radians(Lat2)
    lon2 = radians(Long2)

    dlon = lon2 - lon1    # take differences
    dlat = lat2 - lat1

    ## Haversine Formula (a two-argument, inverse-tangent function; calculates great c
    a = sin(dlat / 2)**2 + cos(lat1) * cos(lat2) * sin(dlon / 2)**2
    c = 2 * atan2(sqrt(a), sqrt(1 - a))
    distanceMI = R * c
    distanceFT = distanceMI * 5280    # convert miles to feet

    return distanceFT
```

In [5]:
```
# FYI test...↔
```

```
Distance between coordinates at sea level:    736.7723999852386
Change with ground elevation of 200 ft:         0.00704732928784324
Change with subsurface elevation of -10000 ft: -0.3523664643869324
```

3 Calculate Azimuth from Two Points (Lat-Long)

In [6]:
```
def azimuth(Lat1, Long1, Lat2, Long2):
    '''Provides azimuth (degrees) between two points (coordinates in degrees of latitu
    lat1 = radians(Lat1)
    lat2 = radians(Lat2)
    dLong = radians(Long2 - Long1)

    x = sin(dLong) * cos(lat2)
    y = cos(lat1) * sin(lat2) - (sin(lat1) * cos(lat2) * cos(dLong))

    angle_radians = atan2(x, y)    # atan2 return values from -π to +π
    angle_degrees = degrees(angle_radians) # convert to -180° to + 180°
    azimuth = (angle_degrees + 360) % 360    # convert to azimuth bearing

    return azimuth
```

In [7]:
```
azimuth(Lat1, Long1, Lat2, Long2)
```

Out[7]:
```
184.31068745872224
```