In [1]:

```
import numpy as np
import pandas as pd
import xarray as xr
import netCDF4
from netCDF4 import Dataset,num2date
import matplotlib as mpl
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
from matplotlib import pyplot as plt
%matplotlib inline
```

1. Global methane levels from 2002

1.1 [5 points] Compute methane climatology for each month, and plot your results in 12 panels.

In [2]:

```
ds = xr.open_dataset("200301_202006-C3S-L3_GHG-PRODUCTS-OBS4MIPS-MERGED-v4.3.nc")
ds
```

Out[2]:

xarray.Dataset

▶ Dimensions: (**time**: 210, bnds: 2, **lat**: 36, **lon**: 72, pressure: 10)

▼ Coordinates:

time	(time)	datetime64[ns]	2003-01-16T12:00:…
lat	(lat)	float64	-87.5 -82.5 -77.5 ……
lon	(lon)	float64	-177.5 -172.5 … 17…

▼ Data variables:

time_bnds	(time, bnds)	datetime64[ns]	…
lat_bnds	(lat, bnds)	float64	…
lon_bnds	(lon, bnds)	float64	…
pre	(pressure)	float64	…
pre_bnds	(pressure, bnds)	float64	…
land_fraction	(lat, lon)	float64	…
xch4	(time, lat, lon)	float32	…
xch4_nobs	(time, lat, lon)	float64	…
xch4_stderr	(time, lat, lon)	float32	…
xch4_stddev	(time, lat, lon)	float32	…
column_averagi…	(time, pressure, lat, lon)	float32	…
vmr_profile_ch4…	(time, pressure, lat, lon)	float32	…

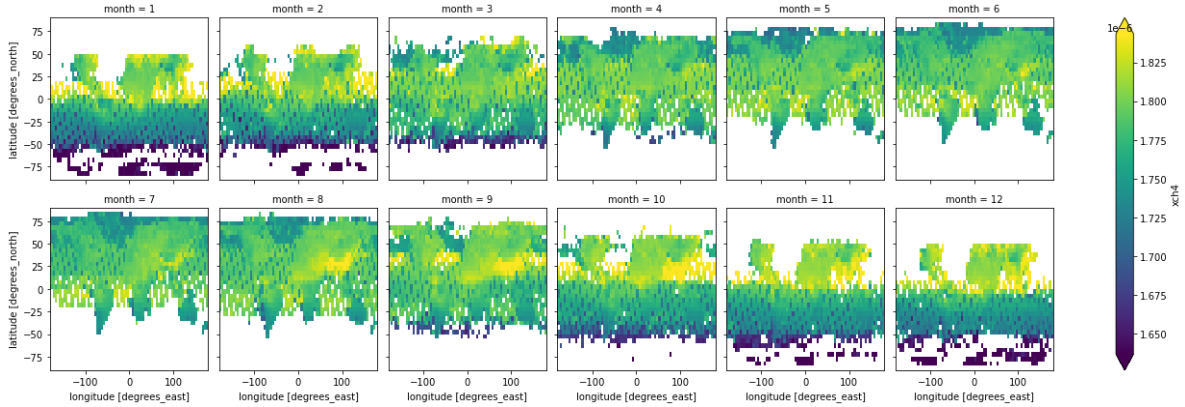
▶ Attributes: (28)

In [3]:

```
group_data=ds.xch4.groupby(ds.time.dt.month).mean()
group_data.plot(col="month",col_wrap=6,robust=True)
```

Out[3]:

<xarray.plot.facetgrid.FacetGrid at 0x1dd610b2790>



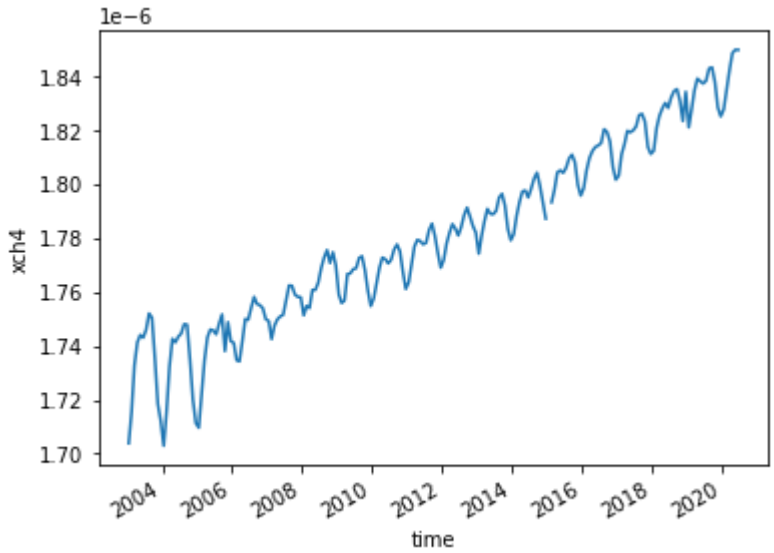
1.2 [5 points] Plot globally-averaged methane from 2003-01 to 2020-06 as a time series. Describe your results. Check your plot with this one.

In [4]:

```
ds.xch4.sel(time=slice('2003-01','2020-06')).mean(dim=['lat','lon']).plot()
```

Out[4]:

[<matplotlib.lines.Line2D at 0x1dd623c9700>]



从图中可以看出，2003-2006甲烷量很高，到2006年甲烷含量急剧下降，到2008年4月份甲烷含量还是上升，在2005年和2009年有一个急剧下降和上升的点，即有一个低值。每年的甲烷量都有都有一个周期的变化，表现为秋冬季节低，夏季高。

1.3 [5 points] Plot deseasonalized methane levels at point [15°S, 150°W] from 2003-01 to 2020-06 as a time series. Describe your results.

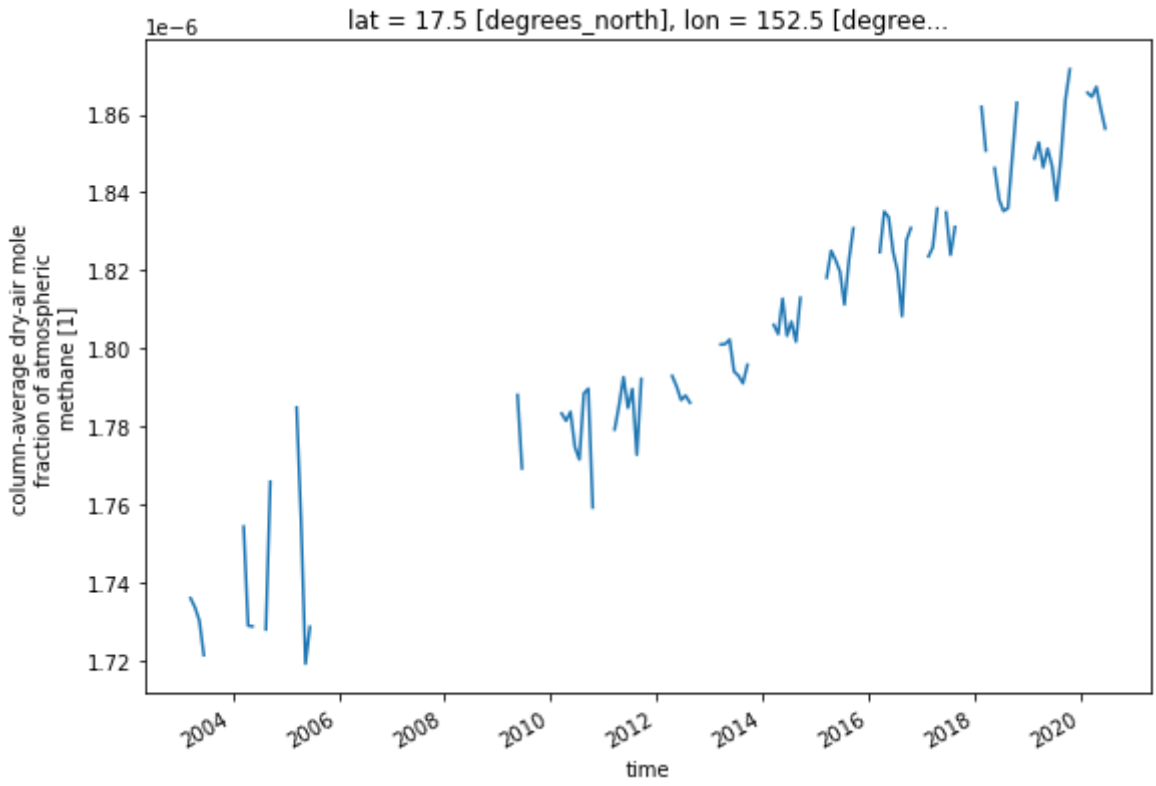
In [5]:

```
group_location=ds.xch4.sel(time=slice('2003-01','2020-06')).sel(lon=150, lat=15, method='nearest')
group_location.plot(size=6)
#
```

```
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
```

Out[5]:

```
[<matplotlib.lines.Line2D at 0x1dd6234f6a0>]
```



从图中可以看出，在这个位置上,有很多缺测值，但总体仍然呈现出上升趋势。

2. Niño 3.4 index

The Niño 3.4 anomalies may be thought of as representing the average equatorial sea surface temperatures (SSTs) across the Pacific from about the dateline to the South American coast (5N-5S, 170W-120W). The Niño 3.4 index typically uses a 3-month running mean, and El Niño or La Niña events are defined when the Niño 3.4 SSTs exceed +/- 0.5°C for a period of 5 months or more. Check Equatorial Pacific Sea Surface Temperatures for more about the Niño 3.4 index.

In [6]:

```
ds = xr.open_dataset('NOAA_NCDC_ERSST_v3b_SST.nc')
ds
```

Out[6]:

xarray.Dataset

▶ Dimensions: (**lat**: 89, **lon**: 180, **time**: 684)

▼ Coordinates:

lat	(lat)	float32	-88.0 -86.0 -84.0 ... 86.0 88.0	
lon	(lon)	float32	0.0 2.0 4.0 ... 354.0 356.0 35…	
time	(time)	datetime64[ns]	1960-01-15 ... 2016-12-15	

▼ Data variables:

sst	(time, lat, lon)	float32	...

▼ Attributes:

Conventions :	IRIDL
source :	https://iridl.ldeo.columbia.edu/SOURCES/.NOAA/.NCDC/.ERSST/.version3b/.sst/
history :	extracted and cleaned by Ryan Abernathey for Research Computing in Earth Science

1.1 [10 points] Compute monthly climatology for SST from Niño 3.4 region, and subtract climatology from SST time series to obtain anomalies.

In [7]:

```
#选择尼诺3.4区域的sst数据
Nino_34=ds.sst.sel(lat=slice(-5,5),lon=slice(190,240))
```

In [8]:

```
anomalies=Nino_34.groupby(ds.time.dt.month)-Nino_34.groupby(ds.time.dt.month).mean(dim='time')
resample_obj = anomalies.resample(time="3M")
ds_anom_resample = resample_obj.mean(dim="time")
ds_anom_resample
```

Out[8]:

xarray.DataArray 'sst' (**time**: 229, **lat**: 5, **lon**: 26)

```
array([[[-4.31577682e-01, -4.18462753e-01, -3.97953033e-01, ...,
         -2.11664200e-01, -2.37762451e-01, -2.44014740e-01],
        [-4.12590027e-01, -4.06719208e-01, -3.87514114e-01, ...,
         -5.20648956e-01, -5.34645081e-01, -5.19971848e-01],
        [-4.09322739e-01, -3.97438049e-01, -3.62377167e-01, ...,
         -6.37388229e-01, -6.17195129e-01, -5.83724976e-01],
        [-4.14085388e-01, -3.79093170e-01, -3.21561813e-01, ...,
         -4.32926178e-01, -3.84042740e-01, -3.35262299e-01],
        [-5.04367828e-01, -4.38940048e-01, -3.71025085e-01, ...,
         -1.74535751e-01, -1.10445023e-01, -6.91814423e-02]],

       [[-4.34304565e-01, -4.11919922e-01, -3.80833298e-01, ...,
         -2.34196976e-01, -2.69255310e-01, -2.73969650e-01],
        [-4.45957184e-01, -4.12000030e-01, -3.67994308e-01, ...,
         -5.37923813e-01, -5.70958436e-01, -5.63295364e-01],
        [-4.64090973e-01, -4.16372925e-01, -3.52096558e-01, ...,
         -6.36102021e-01, -6.32293701e-01, -6.25462830e-01],
        [-4.85488266e-01, -4.30688858e-01, -3.54335785e-01, ...,
         -4.52582687e-01, -4.23582703e-01, -4.12778229e-01],
        [-5.82661331e-01, -5.20365417e-01, -4.54551697e-01, ...,
...
        [-7.30498016e-01, -8.12325180e-01, -8.80907059e-01, ...,
         -7.10569382e-01, -6.31128311e-01, -5.32169998e-01],
        [-9.77057755e-01, -1.03931940e+00, -1.08908331e+00, ...,
         -7.64347076e-01, -6.85469329e-01, -6.09845459e-01],
        [-7.90389359e-01, -8.42882812e-01, -8.73820007e-01, ...,
         -4.55050796e-01, -4.07323211e-01, -3.71250153e-01],
        [-3.22086334e-01, -3.76546234e-01, -4.13348526e-01, ...,
         -2.10369110e-01, -1.91286087e-01, -1.61457703e-01]],

       [[-4.54637527e-01, -5.11337280e-01, -5.60937881e-01, ...,
         -2.93106079e-01, -2.01011658e-01, -8.88051987e-02],
        [-8.78618240e-01, -9.42394257e-01, -9.98949051e-01, ...,
         -7.94350624e-01, -6.92551613e-01, -5.67633629e-01],
        [-1.07530212e+00, -1.14840794e+00, -1.21267223e+00, ...,
         -9.89422798e-01, -8.89242172e-01, -7.72073746e-01],
        [-9.11746979e-01, -9.98181343e-01, -1.07053089e+00, ...,
         -6.86944962e-01, -6.11206055e-01, -5.13758659e-01],
        [-5.31754494e-01, -6.18759155e-01, -6.96014404e-01, ...,
         -3.98782730e-01, -3.31537247e-01, -2.39914894e-01]]],
      dtype=float32)
```

▼ Coordinates:

time	(time)	datetime64[ns]	1960-01-31 ... 2017-01-31
lat	(lat)	float32	-4.0 -2.0 0.0 2.0 4.0
lon	(lon)	float32	190.0 192.0 194.0 ... 238.0 240.0

▶ Attributes: (0)

1.2 [10 points] Visualize the computed Niño 3.4. Your plot should look similar to this one

In [9]:

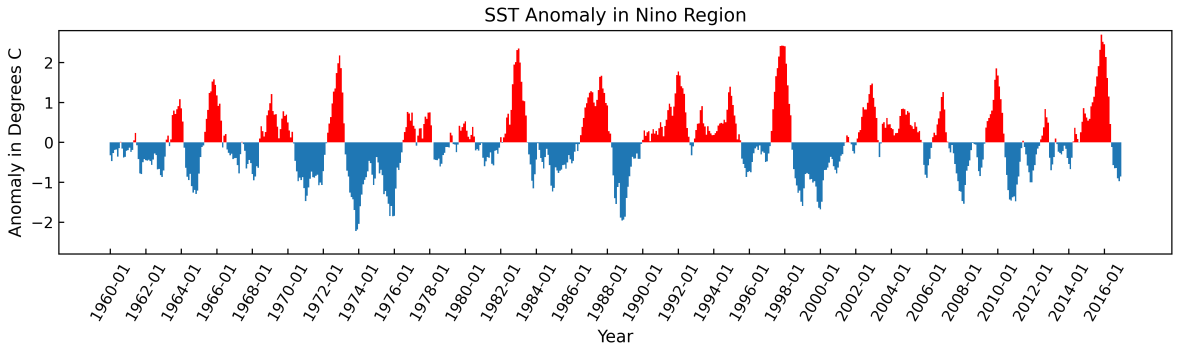
```
## 时间序列是一个二维的，因此要去掉维度，将一时间的所有区域的异常进行平均。
##
month_anomalies=anomalies.mean(dim=['lon','lat'])
time = np.array(month_anomalies['time'].dt.strftime('%Y-%m'))#取出数据的时间序列作为x轴
```

In [10]:

```
fig=plt.figure(figsize=(10,2),dpi=500)#添加画布
ax1=fig.add_axes([0,0,1,1])#添加子图)
# 设置距平正值为红色，负值为蓝色
ax1.bar(time,month_anomalies,color=np.where(month_anomalies>0,'red', 'tab:blue'),width=1)
ax1.set_ylim(-2.8,2.8)#设置y轴的范围
ax1.tick_params(axis='both',which='both',direction='in') #设置刻度线在图内显示
plt.xticks(range(0,len(time),24),rotation=60)  #设置x轴的间隔
ax1.set_title('SST Anomaly in Nino Region')#设置标题
ax1.set_xlabel('Year',fontsize=11)#设置x轴的标签
ax1.set_ylabel('Anomaly in Degrees C',fontsize=11)#设置y轴的标签
```

Out[10]:

```
Text(0, 0.5, 'Anomaly in Degrees C')
```



3. Explore a netCDF dataset

In [11]:

```
## 地表两米的空气温度，月平均，lat=[88.542,-88.542],lon=[0,358.125] 时间：1979年1月-2022年10月
ds = xr.open_dataset('air.2m.mon.mean.nc')
ds
```

Out[11]:

xarray.Dataset

▶ Dimensions: (**level**: 1, **lat**: 94, **lon**: 192, **time**: 526, nbnds: 2)

▼ Coordinates:

level	(level)	float32	2.0
lat	(lat)	float32	88.54 86.65 84.75 ... -8…
lon	(lon)	float32	0.0 1.875 3.75 ... 356.2…
time	(time)	datetime64[ns]	1979-01-01 ... 2022-10…

▼ Data variables:

time_bnds	(time, nbnds)	datetime64[ns]	...
air	(time, level, lat, lon)	float32	...

▶ Attributes: (10)

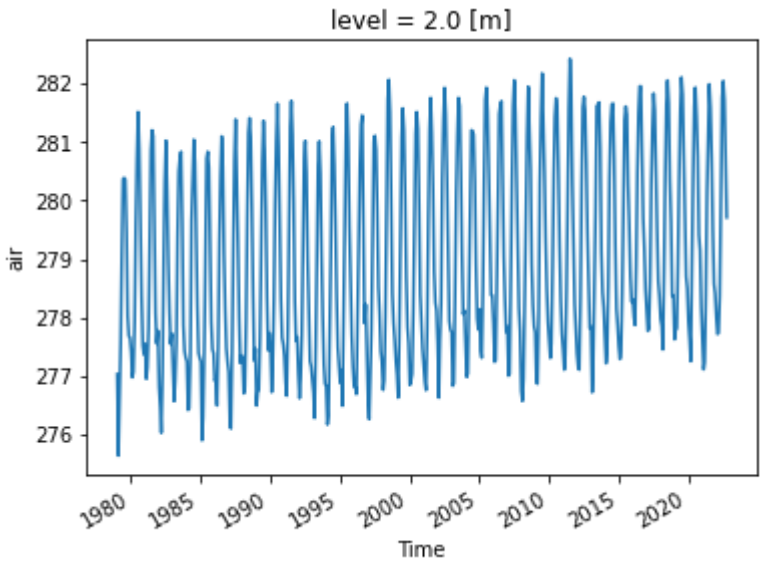
3.1 [5 points] Plot a time series of a certain variable with monthly seasonal cycle removed.

In [12]:

```
series_year=ds.air.groupby(ds.time.dt.year).mean(dim=['lat','lon'])
series_year.plot()
#地面两米的空气温度，有一个年际周期性变化。并且总体有一个上升的趋势（不太明显）
```

Out[12]:

```
[<matplotlib.lines.Line2D at 0x1dd6356b130>]
```



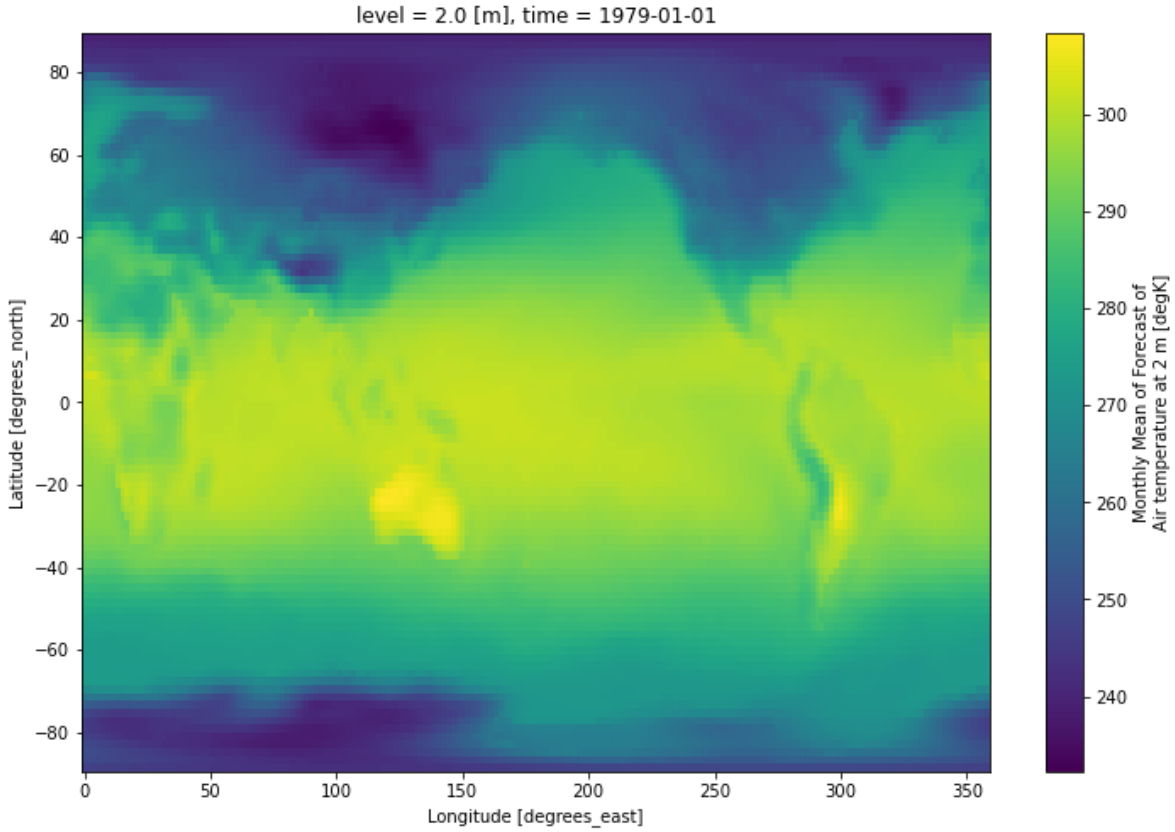
3.2 [10 points] Make at least 5 different plots using the dataset.

In [13]:

```
#1 1979年1月份的年全球地表2米的温度情况情况
ds.air.isel(time=0).plot(size=8)
##可以看出，两极的温度低，澳大利亚以及南美的西海岸温度较高
```

Out[13]:

<matplotlib.collections.QuadMesh at 0x1dd64ac17f0>



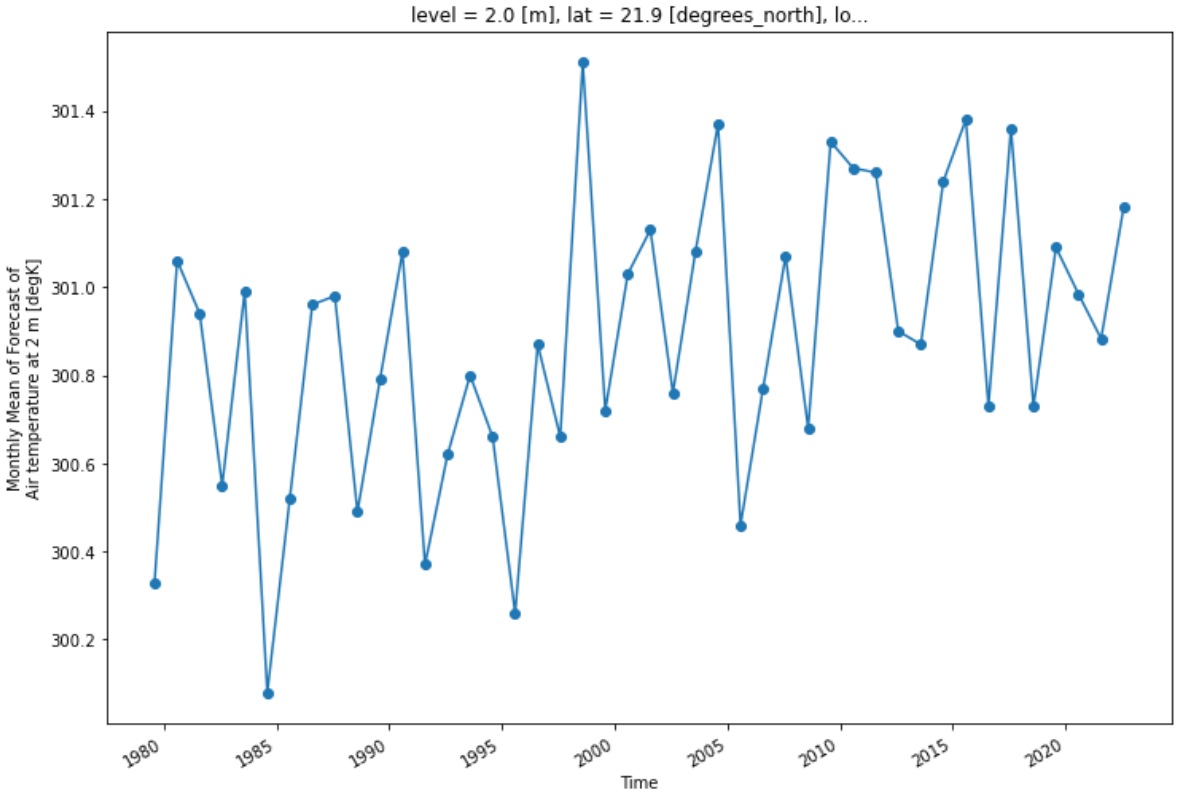
In [14]:

```
#画深圳的 2月的的地面两米的气温数据的从1854-2021年的时间序列
ds.air.groupby(ds.time.dt.month)[8].sel(lon=114.06, lat=22.54,method='nearest').plot(marker="o", si
```

```
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
```

Out[14]:

```
[<matplotlib.lines.Line2D at 0x1dd64b74ee0>]
```

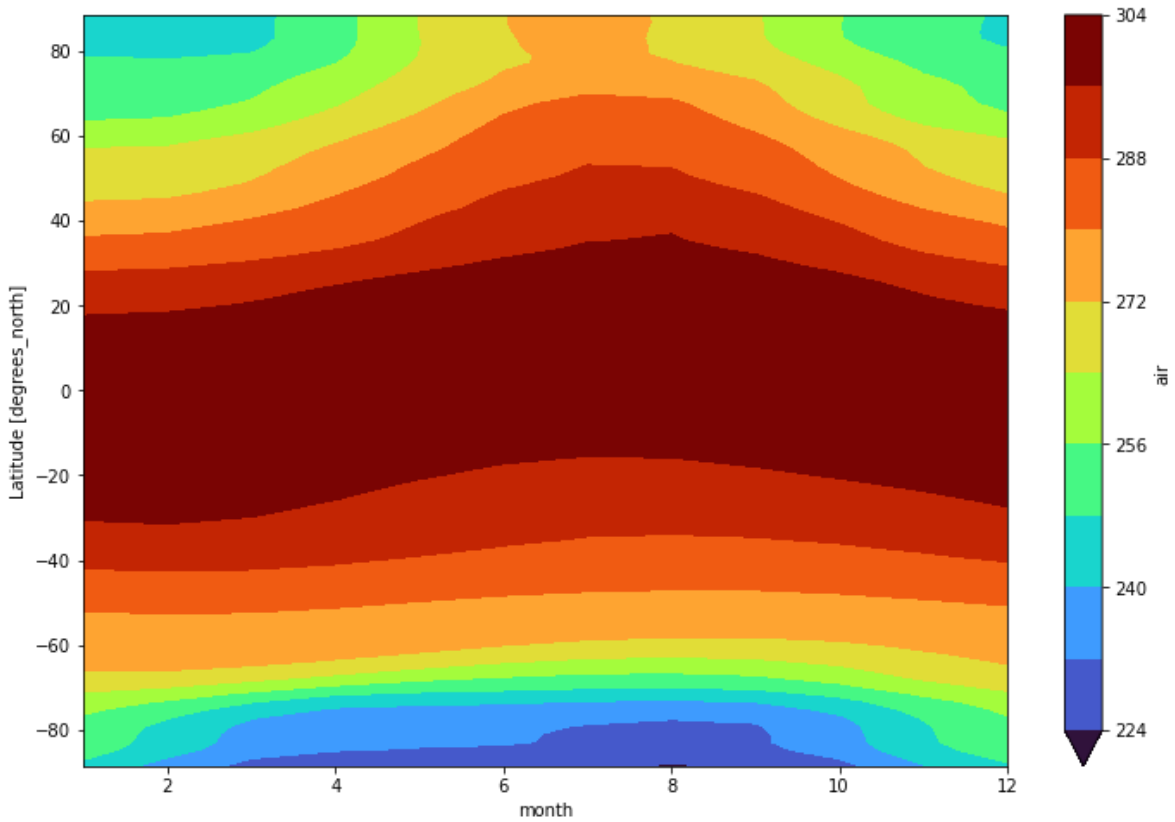


In [15]:

```
# Plot mean climatology
tas_clim = ds.air.groupby('time.month').mean()
tas_clim.mean(dim=['lon','level']).transpose().plot.contourf(levels=12, robust=True, cmap='turbo',s
```

Out[15]:

<matplotlib.contour.QuadContourSet at 0x1dd64be3bb0>



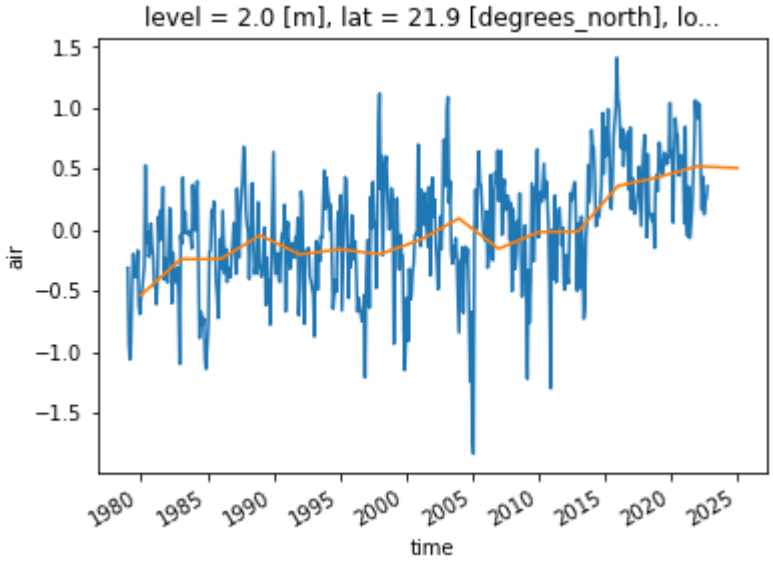
In [16]:

```
#4

# Group data by month
group_data = ds.air.groupby('time.month')

# Apply mean to grouped data, and then compute the anomalies
air_anom = group_data - group_data.mean(dim='time')
air_anom

# Use resample() function at a frequency of 3 years
resample_obj = air_anom.resample(time="3Y")

# Show the resample object
resample_obj

# Apply mean() function to the resample object and get results
ds_anom_resample = resample_obj.mean(dim="time")
ds_anom_resample

# Plot anomalies
air_anom.sel(lon=114.55+180, lat=22.5,
             method='nearest').plot()

# Plot 3-year averaged anomalies
ds_anom_resample.sel(lon=114.55+180, lat=22.5,
                     method='nearest').plot()
```

```
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
```

Out[16]:

```
[<matplotlib.lines.Line2D at 0x1dd68164ee0>]
```


level = 2.0 [m], lat = 21.9 [degrees_north], lo...
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
air
1980
1985
1990
1995
2000
2005
2010
2015
2020
2025
time

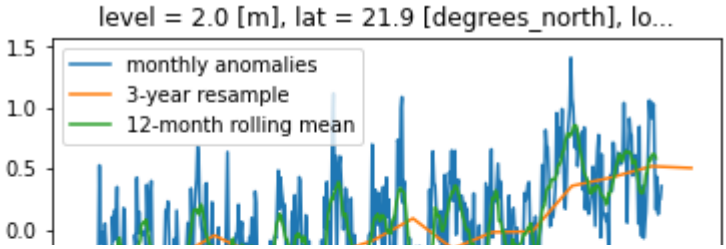
In [17]:

```
# Compute rolling means
ds_anom_rolling = air_anom.rolling(time=12, center=True).mean()

# Show rolling means
ds_anom_rolling

# Plot anomalies
air_anom.sel(lon=114.55+180, lat=22.5, method='nearest').plot(
    label="monthly anomalies")

# Plot 3-year averaged anomalies
ds_anom_resample.sel(lon=114.55+180, lat=22.5, method='nearest').plot(
    label="3-year resample")

# Plot 12-month rolling mean
ds_anom_rolling.sel(lon=114.55+180, lat=22.5, method='nearest').plot(
    label="12-month rolling mean")

# Add the legend
plt.legend()
```

```
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
C:\Users\nicol\anaconda3\lib\site-packages\xarray\core\indexes.py:234: FutureWarnin
g: Passing method to Float64Index.get_loc is deprecated and will raise in a future v
ersion. Use index.get_indexer([item], method=...) instead.
  indexer = self.index.get_loc(
```

Out[17]:

```
<matplotlib.legend.Legend at 0x1dd6818a8e0>
```



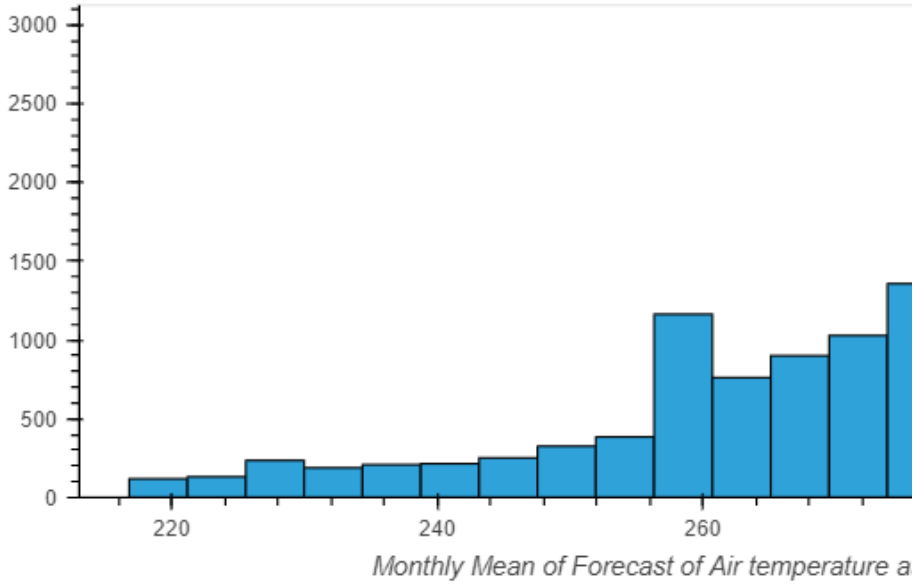
In [18]:

```
import hvplot.xarray
```

In [19]:

```
ds.air.isel(time=-1).hvplot(cmap="fire")
```

Out[19]:



In [20]:

```
ds.air.hvplot(groupby="time", clim=(ds.air.min(), ds.air.max()), cmap='turbo')
```

Out[20]: