Quellcode Aktienkurs Rechner

MainActivity.java

```
public class MainActivity extends AppCompatActivity implements
AdapterView.OnItemSelectedListener{

    private String apiURL="https://www.alphavantage.co/query";

    private ProgressBar progressBar;
    private EditText symbol, zeit;
    private String symbolS;
    private int zeitI;
    private String einheit;

    private Spinner spinner;
    List<Aktie> aktieList;
    private TextView ausgabe;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        progressBar = findViewById(R.id.progressBar);
        spinner = (Spinner) findViewById(R.id.spinner);
        symbol = (EditText) findViewById(R.id.symbol);
        zeit = (EditText) findViewById(R.id.zeit);
        ausgabe = (TextView) findViewById(R.id.ausgabe);
        ArrayAdapter<CharSequence> adapter =
ArrayAdapter.createFromResource(this,R.array.time_array,android.R.layout.simple_sp
inner_item);

adapter.setDropDownViewResource(R.layout.support_simple_spinner_dropdown_item);
        spinner.setAdapter(adapter);
        spinner.setOnItemSelectedListener(this);
        einheit = spinner.getItemAtPosition(0).toString();

        progressBar.setVisibility(View.INVISIBLE);
    }

    public void rechne(View view) {
        ausgabe.setTextColor(Color.WHITE);

        apiURL="https://www.alphavantage.co/query";
        ausgabe.setText("");
        aktieList = new ArrayList<>();

        symbolS = symbol.getEditableText().toString();
```

```
        if(symbolS.equals("")){
            ausgabe.setText("Bitte geben Sie Tickersymbol und Zeitpunkt ein");
            ausgabe.setTextColor(Color.RED);
            return;
        }
        try {
            zeitI = Integer.parseInt(zeit.getEditableText().toString());
        }
        catch (Exception e){
            ausgabe.setText("Bitte überprüfen Sie die angegebene Zeit");

            ausgabe.setTextColor(Color.RED);
            return;
        }




        String time="";

        if(einheit.equals("Tag/e")){
            time="function=TIME_SERIES_DAILY";
        }else if(einheit.equals("Woche/n")){
            time = "function=TIME_SERIES_WEEKLY";
        }else if(einheit.equals("Monat/e")){
            time="function=TIME_SERIES_MONTHLY";
        }

        apiURL += "?"+time+"&symbol="+symbolS+"&apikey=Y2LQAW31V2HXBEJL";

        GetApiData getApiData = new GetApiData();
        getApiData.execute();
    }

    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int position,
long id) {
        this.einheit = parent.getItemAtPosition(position).toString();
    }

    @Override
    public void onNothingSelected(AdapterView<?> parent) {

    }


    private class GetApiData extends AsyncTask<Void,Void,Void>{

        @Override
        protected void onPreExecute() {
            progressBar.setVisibility(View.VISIBLE);
        }

        @Override
```

```
        protected Void doInBackground(Void... voids) {

            try {
                URL myurl = new URL(apiURL);
                HttpURLConnection httpURLConnection =
(HttpURLConnection)myurl.openConnection();
                httpURLConnection.setDoInput(true);
                httpURLConnection.setRequestMethod("GET");
                httpURLConnection.connect();


                InputStream is = httpURLConnection.getInputStream();
                BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(is));
                String line ="";

                StringBuilder stringBuilder = new StringBuilder();
                while ((line = bufferedReader.readLine())!=null){
                    stringBuilder.append(line);
                }
                String data = stringBuilder.toString();



                JSONObject parentObject = new JSONObject(data);
                JSONObject time = null;
                Iterator<String> keyTimes = parentObject.keys();
                while (keyTimes.hasNext()){
                    String keyT = keyTimes.next();
                    if(keyT.equals("Meta Data")){
                        keyT = keyTimes.next();
                    }
                    time = parentObject.getJSONObject(keyT);
                }



                Iterator<String> keys = time.keys();
                while (keys.hasNext()){
                    Aktie aktie = new Aktie();
                    String key = keys.next();
                    JSONObject obj = time.getJSONObject(key);
                    aktie.setDatum(key);
                    aktie.setClose(obj.getDouble("4. close"));
                    aktie.setOpen(obj.getDouble("1. open"));
                    double rendite = Math.log(aktie.getClose()/aktie.getOpen());
                    aktie.setRendite(rendite);
                    aktieList.add(aktie);
                }
            }
            catch (Exception e){
                e.printStackTrace();
            }
            return null;
```

```
        }

        @Override
        protected void onPostExecute(Void aVoid) {
            progressBar.setVisibility(View.INVISIBLE);
            String out="";
            int n=0;
            double mittelwert;
            double rendite = 0;
            for(Aktie aktie: aktieList){
                n++;
                rendite += aktie.getRendite();
            }
            if(n>0){
                mittelwert = Math.round(rendite/n *1000000.0) / 1000000.0;

                double zaehler = 0;
                for (Aktie aktie: aktieList){
                    zaehler += Math.pow((aktie.getRendite()-mittelwert),2);
                }

                double sa = Math.round(Math.sqrt(zaehler/n) *1000000.0) /
1000000.0;

                double [] werte = new double[zeitI+1];
                double [] werteLast = new double[zeitI+1];

                werteLast[0] = aktieList.get(0).getClose();

                int c=0;

                while(c<zeitI) {

                    for (int k = 0; k < c + 2; k++) {
                        if (k == 0) {
                            werte[k] = berechneAufstieg(werteLast[k], mittelwert,
sa);
                        } else {
                            werte[k] = berechneAbstieg(werteLast[k - 1],
mittelwert, sa);
                        }
                    }
                    for (int k = 0; k < c + 2; k++) {
                        werteLast[k] = werte[k];
                    }
                    c++;
                }

                for(int i=0; i<werte.length; i++){
                    werte[i] = Math.round(werte[i] * 100.0) / 100.0;
                }

                //Date currentTime = Calendar.getInstance().getTime();
                SimpleDateFormat sdf = new SimpleDateFormat("dd.MM.yyyy",
```

```
Locale.getDefault());
                String date = sdf.format(new Date());

                SimpleDateFormat sdf1 = new SimpleDateFormat("yyyy-MM-dd");
                String dt = sdf1.format(new Date());

                Calendar c1 = Calendar.getInstance();
                try {
                    c1.setTime(sdf1.parse(dt));
                } catch (ParseException e) {
                    e.printStackTrace();
                }
                if(einheit.equals("Tag/e")){
                    c1.add(Calendar.DATE, zeitI);
                }else if(einheit.equals("Woche/n")){
                    c1.add(Calendar.WEEK_OF_YEAR, zeitI);
                }
                else if(einheit.equals("Monat/e")){
                    c1.add(Calendar.MONTH, zeitI);
                }
                dt = sdf.format(c1.getTime());



                double akt = Math.round(aktieList.get(0).getClose() * 100.0)
/100.0;

                out = "Aktueller Wert("+date+"): "+akt+"$\n\n";
                out += "Prognostzitierte Werte am "+dt+"\n";
                for(int i = 0; i<werte.length;i++){
                    out +=werte[i]+"$\n";
                }
                //out="mittelwert: "+mittelwert+" SA: "+sa+"\nAktuller Wert:
"+akt+"$\n1. Aufstieg: "+aufstieg+"$\n1.Abstieg: "+abstieg+"$";
            }

            if(aktieList.isEmpty()) out = "Bitte Internetverbindung oder
Tickersymbol überprüfen!";
            ausgabe.setText(out);
        }

        private double berechneAufstieg(double akt,double mittelwert, double sa){
            return Math.pow(Math.E,mittelwert+sa)*akt;
        }

        private double berechneAbstieg(double akt,double mittelwert, double sa){
            return Math.pow(Math.E,mittelwert-sa)*akt;
        }
    }
}
```

Aktie.java

```
public class Aktie {

    private String datum;
    private Double open;
    private Double close;
    private Double rendite;

    public String getDatum() {
        return datum;
    }

    public void setDatum(String datum) {
        this.datum = datum;
    }

    public Double getOpen() {
        return open;
    }

    public void setOpen(Double open) {
        this.open = open;
    }

    public Double getClose() {
        return close;
    }

    public void setClose(Double close) {
        this.close = close;
    }

    public Double getRendite() {
        return rendite;
    }

    public void setRendite(Double rendite) {
        this.rendite = rendite;
    }
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:weightSum="10"
        android:layout_marginTop="30dp">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_weight="5">

            <TextView
                android:layout_gravity="center"
                android:paddingTop="20dp"
                android:paddingLeft="20dp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Tickersymbol:"/>

        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_weight="5">

            <EditText
                android:id="@+id/symbol"
                android:paddingLeft="20dp"
                android:layout_width="120dp"
                android:layout_height="50dp"/>

        </LinearLayout>

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:weightSum="10"
        android:layout_marginTop="30dp">

        <LinearLayout
            android:layout_width="match_parent"
```

```
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_weight="5">

            <TextView
                android:paddingTop="20dp"
                android:paddingLeft="20dp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Wie viel ist meine Aktie Wert in:"/>

        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_weight="5">

            <EditText
                android:id="@+id/zeit"
                android:paddingLeft="20dp"
                android:layout_width="60dp"
                android:layout_height="50dp"/>

            <Spinner
                android:id="@+id/spinner"
                android:entries="@array/time_array"
                android:layout_gravity="center"
                android:layout_width="135dp"
                android:layout_height="wrap_content"/>

        </LinearLayout>

    </LinearLayout>


    <LinearLayout
        android:layout_marginTop="20dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <Button
            style="@style/Widget.AppCompat.Button.Colored"
            android:onClick="rechne"
            android:layout_width="250dp"
            android:layout_height="50dp"
            android:text="Jetzt Berechnen"
            android:layout_gravity="center"/>

    </LinearLayout>

    <androidx.cardview.widget.CardView
```

```
            android:layout_marginLeft="20dp"
            android:layout_marginRight="20dp"
            android:layout_marginTop="30dp"
            android:layout_width="match_parent"
            android:layout_height="400dp"
            app:cardCornerRadius="30dp"
            app:cardBackgroundColor="@color/cardview_dark_background">


            <ScrollView
                android:layout_width="match_parent"
                android:layout_height="match_parent">


                <LinearLayout
                    android:layout_width="match_parent"
                    android:layout_height="match_parent"
                    android:orientation="vertical">

                    <TextView
                        android:id="@+id/ausgabe"
                        android:layout_gravity="center"
                        android:textColor="#FFFFFF"
                        android:padding="10dp"
                        android:layout_width="match_parent"
                        android:layout_height="match_parent"
                        android:text="" />

                    <ProgressBar
                        android:id="@+id/progressBar"
                        style="?android:attr/progressBarStyle"
                        android:layout_width="match_parent"
                        android:layout_height="wrap_content" />

                </LinearLayout>

            </ScrollView>




        </androidx.cardview.widget.CardView>

    </LinearLayout>



</RelativeLayout>
```