

```

/**
 * @file      Bestellung.h
 * @synopsis   Deklaration Klasse Bestellung
 * @author    Jan Tammen (FH Konstanz), <jan.tammen@fh-konstanz.de>
 * @author    Christoph Eck (FH Konstanz), <christoph.eck@fh-konstanz.de>
 * @date      2005-05-10
 */

#ifndef BESTELLUNG_H
#define BESTELLUNG_H

#include "Daten.h"
#include "Position.h"

#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <functional>
using namespace std;

class Bestellung
{
public:
    Bestellung ();
    ~Bestellung ();

    void addPosition (const Position& pos);
    double getGesamtpreis () const { return mGesamtpreis; }

    void init ();
    void displayAllPos () const;
    void displayPosWithTyp (const int typ) const;
    friend ostream& operator<< (ostream&, const Bestellung&);

private:
    struct Zeitpunkt
    {
        string datum;
        string uhrzeit;
    };

    Zeitpunkt mDateTime;
    int mAnzahlArtikel;
    double mGesamtpreis;
    vector< Position > mPositionen;

    /// Praedikat fuer Suche nach best. Positionen
    struct DifferentTyp :
        public binary_function<Position, Artikelliste::Artikeltyp, bool>
    {
        bool operator() (const Position& p,
                        const Artikelliste::Artikeltyp& typ) const
        {
            return p.getArtikeltyp() != typ;
        }
    };

};

inline ostream& operator<< (ostream& out, const Bestellung& x)
{
    out << x.mDateTime.datum << ", " << x.mDateTime.uhrzeit << " Uhr";
    return out;
}

#endif

```