

Обработка данных в формате  
дата/время. Вывод данных из  
программы на печать.

# План темы:

- 1. Тип TDateTime.
- 2. Операции с данными типа TDateTime.
- 3. Процедуры и Функции для данных типа TDateTime.
- 4. Компоненты MonthCalendar, DateTimePicker, Timer.
- 5. Вывод данных из программы на печать.
- 6. Пример программы.

# 1. Тип TDateTime.

- Для хранения и обработки значений даты и времени в Delphi применяется тип данных **TDateTime**.
- Он соответствует типу **Double** (вещественный с двойной точностью).
- Целая часть элемента данных типа TDateTime соответствует количеству дней, прошедших с полночи 30 декабря 1899 года.
- Дробная часть соответствует времени дня.

## 2. Операции с данными типа TDateTime.

- $\text{Дата\_2} - \text{Дата\_1} = \langle \text{Кол-во суток между датами} \rangle;$
- $\text{Дата\_1} + \langle \text{Кол-во суток} \rangle = \text{Дата\_2};$
- $\text{Дата\_2} - \langle \text{Кол-во суток} \rangle = \text{Дата\_1};$
- **Формат записи константы TDateTime:**  
Число.Месяц.Год Часы:Минуты:Секунды

### 3. Процедуры и функции для данных типа TDateTime.

#### Функции:

- **Date** – возвращает текущую дату.
- **Time** – возвращает текущее время.
- **Now** – возвращает текущие дату и время.
- **DayOfWeek(D)** – возвращает номер дня недели даты D (1 – воскресенье, 2 – понедельник, ..., 7 – суббота).

### 3. Процедуры и функции для данных типа TDateTime.

- **DateTimeToStr (D)** – преобразует дату и время D в строку.
- **StrToDateTime (S)** – преобразует строку S в дату и время.
- **DateToStr (D)** – преобразует дату D в строку.
- **StrToDate (S)** – преобразует строку S в дату.
- **TimeToStr (D)** – преобразует время D в строку.
- **StrToTime (S)** – преобразует строку S в время.

### 3. Процедуры и функции для данных типа TDateTime.

- **EncodeDate (g, m, d)** – возвращает дату, сформированную по году (g), месяцу (m), дню (d).
- **EncodeTime (ch, min, sec, msec)** – возвращает время, сформированное по часам (ch), минутам (min), секундам (sec), миллисекундам (msec).

### 3. Процедуры и функции для данных типа TDateTime.

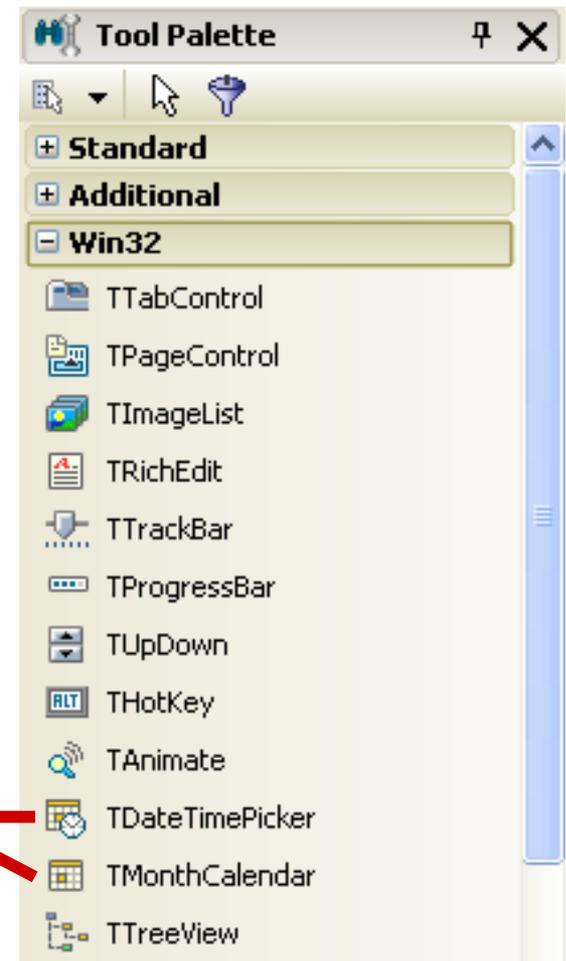
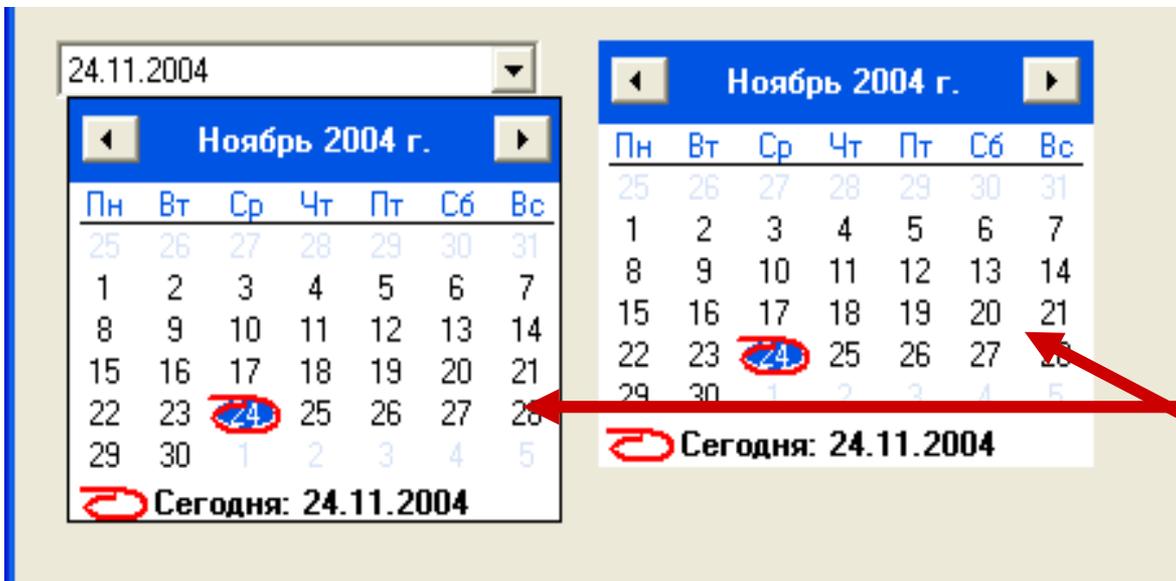
#### Процедуры:

- **DecodeDate (D, g, m, d)** – Разбивает значение даты D на год (g), месяц (m), день (d).
- **DecodeTime (T, ch, min, sec, msec)** – Разбивает значение времени T на часы (ch), минуты (min), секунды (sec), миллисекунды (msec).

Переменные g, m, d, ch, min, sec, msec должны иметь тип **Word** (целый беззнаковый).

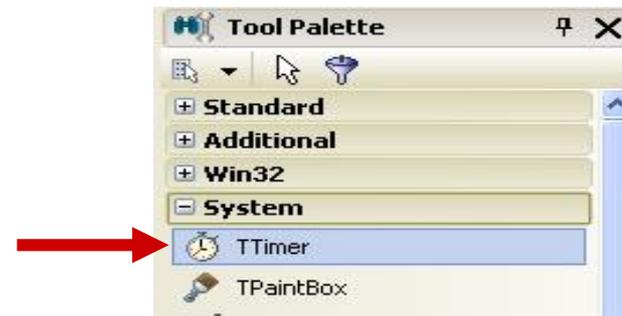
# 4. Компоненты MonthCalendar, DateTimePicker, Timer.

- Для ввода и отображения данных в формате дата удобны компоненты **DateTimePicker** или **MonthCalendar**.



## 4. Компоненты MonthCalendar, DateTimePicker, Timer.

- **Таймер** - это системный генератор событий, который периодически сообщает программе о завершении заданного промежутка времени (событие *OnTimer*).
- Интервал времени между таймерными событиями может устанавливаться в диапазоне от 1 до 65535 миллисекунд (свойство *Interval*).



## 5. Вывод данных из программы на печать.

- Некоторые компоненты (**RichEdit, Chart**) имеют специальный метод **Print**, который выводит данные из этих компонентов на печать.
- Для большинства компонентов (**Memo, StringGrid** и др.) необходимо организовать вывод их данных на печать аналогично выводу данных в текстовый файл.

## 5. Вывод данных из программы на печать.

- Для выбора принтера и настройки печати на форму необходимо установить невизуальный компонент **PrintDialog** (раздел **Dialog** в Палитре компонентов).
- Вызов соответствующего окна диалога осуществляет метод **Execute**, который возвращает значение `True`, если пользователь завершил диалог по кнопке `<Ok>`.

## 5. Вывод данных из программы на печать.

- Сформировать очередную строку выводимой информации, например:

```
For i:=1 to n do
```

```
    S := Memo1.Lines[i];
```

- Вывести строку на принтер:

```
WriteLn(Prn, S);
```

- Закрывать принтер:

```
CloseFile(Prn);
```