# Динамично създаване на нов графичен обект

Принципите са идентични за почти всички визуални компоненти във VCL (Visual Component Library) на Borland C++ Builder 6. Независимо дали създавате `TEdit`, `TLabel`, `TImage` или `TPanel`, стъпките винаги следват един и същ модел от 4 стъпки.

Ето „универсалната формула“ за създаване на графичен обект:

### Общият модел за създаване

// 1. Заделяне на памет (Constructor)
**TClassName *MyObject = new TClassName(Owner);**

// 2. Поставяне върху контейнер (Parent) - НАЙ-ВАЖНОТО!
**MyObject->Parent = Form1;**

// 3. Конфигурация (Свойства)
MyObject->Left = 10; // или някакво друго отстояние
MyObject->Top = 10;
MyObject->Caption = "Текст"; // заглавие на вашия обект

// 4. Логика (Events) - Опционално
**MyObject->OnClick = MyFunction;** //какво събитие да се стартира

### Използване на свойствата : Owner и Parent

Това е най-честото объркване при начинаещите. Разликата е ключова:

1. **Owner (Собственик):** Подава се в скобите на `new TButton(this)`.
   - Той отговаря за **паметта**. Когато собственикът (формата) бъде унищожен, той автоматично извиква `delete` за обекта.
2. **Parent (Родител):** Задава се чрез свойството `->Parent = this;`.
   - Той отговаря за **визуализацията**. Ако обектът няма `Parent`, той съществува в RAM паметта, но е невидим.

---

### Сравнителна таблица за различни обекти

| Обект | Клас (Class Name) | Важно свойство | Примерна употреба |
|---|---|---|---|
| **Текстово** | `TEdit` | `Text` | `MyEdit->Text = "Пиши тук";` |

| Обект | Клас (Class Name) | Важно свойство | Примерна употреба |
|---|---|---|---|
| **поле** | | | |
| **Етикет** | `TLabel` | `Caption` | `MyLabel->Caption = "Име:";` |
| **Картинка** | `TImage` | `Picture` | `MyImage->Picture->LoadFromFile("c:\\img.bmp");` |
| **Панел** | `TPanel` | `BevelOuter` | Служи за групиране на други обекти. |
| **Списък** | `TListBox` | `Items` | `MyList->Items->Add("Първи ред");` |

Ако създавате обекти динамично, често е трудно да изчислите точните координати (`Left` и `Top`). Затова често се използва свойството `Align`:

## TPanel *BottomPanel = new TPanel(this);

## BottomPanel->Parent = this;

## BottomPanel->Align = alBottom; // Автоматично се залепя най-долу и се разпъва по ширина

Някои обекти (като `TTimer` или `TOpenDialog`) са **невизуални**. За тях **не се задава** `Parent`, тъй като те не се изчертават върху формата, а само работят в "бекграунда".

Като пример ще  създадем динамичен графичен бутон

## Код за създаване на динамичен бутон

Поставете този код в събитието на някой съществуващ бутон или в `OnCreate` на формата:

```
void __fastcall TForm1::ButtonCreateClick(TObject *Sender)
{
  // 1. Създаваме обекта в паметта
  TButton *MyButton = new TButton(this);

  // 2. Задаваме му "родител" (къде да се появи)
  MyButton->Parent = this; // Може да бъде и Panel1, GroupBox1 и т.н.
```

```
  // 3. Настройваме свойствата му
  MyButton->Caption = "Динамичен бутон";
  MyButton->Left = 50;  // Позиция X
  MyButton->Top = 50;   // Позиция Y
  MyButton->Width = 150;

  // 4. (Опционално) Свързваме го със събитие (функция)
  MyButton->OnClick = DynamicButtonClick;
}
```

За да може този бутон всъщност да "прави нещо", когато бъде натиснат, трябва ръчно да дефинирате функция, която да се изпълнява.

1. **В заглавния файл (`Unit1.h`)**, в секцията `published` или `private`, добавете декларацията:

```
void __fastcall DynamicButtonClick(TObject *Sender);
```

**В основния файл (`Unit1.cpp`)**, напишете самата логика:

```
void __fastcall TForm1::DynamicButtonClick(TObject *Sender)
{
  ShowMessage("Натиснахте динамично създадения бутон!");
}
```

**Свойството `Parent`:** Това е най-честата грешка. Ако създадете бутон с `new`, но не му присвоите `Parent`, той ще съществува в паметта, но **няма да се вижда** на екрана.

**Управление на паметта:** Тъй като задаваме `this` (формата) като `Owner` в конструктора `new TButton(this)`, C++ Builder автоматично ще изтрие бутона от паметта, когато затворите формата. Не е нужно да викате `delete` ръчно, освен ако не искате да го премахнете по-рано.

**Координати:** Ако създавате няколко бутона с един и същ код, те ще се застъпят (ще са един върху друг). Можете да използвате променлива, за да променяте `Top` позицията при всяко кликване.