) Для чего в WPF используются ресурсы? Каким образом можно определить новый ресурс и управлять им? Опишите назначение класса ResourceDictionary. Зачем каждый элемент имеет собственную коллекцию ресурсов?

Ресурсы позволяют переносить низкоуровневые детали форматирования (вроде размеров шрифтов) в центральное место, где их легко изменять

Опишите назначение класса ResourceDictionar Предоставляет реализацию хэш-таблицы или словаря, содержащую ресурсы WPF, которые используются компонентами и другими элементами приложения WPF.

2) Какая разница между статическими и динамическими ресурсами?

3) Что такое триггеры ? Для чего в WPF используются триггеры? Назовите основные типы триггеров.

Тригерры позволяют декларативно задать некоторые действия, которые выполняются при изменении свойств стиля. Существует три вида триггеров:

Триггеры свойств: вызываются в ответ на изменения свойствами зависимостей своего значения

Триггеры данных: вызываются в ответ на изменения значений любых свойств (они необязательно должны быть свойствами зависимостей)

Триггеры событий: вызываются в ответ на генерацию событий

Мультитриггеры: вызываются при выполнении ряда условий

4) Что такое локализация и как ее обеспечить.

Локализация — это перевод ресурсов приложения в локализованные версии для конкретных языков и региональных параметров, которые поддерживает приложение.

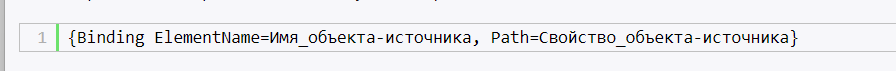
5) Что такое тема? Опишите процесс создания темы на основе ресурсов и стилей.

6) Что такое шаблон и как его создать?

7) Зачем нужны пользовательские элементы управления? Как создать собственный элемент. Опишите члены класса UserControl.

8) Что такое привязка данных? В чем разница между следующими режимами привязки: OneWay, TwoWay и OneTime.

Привязка подразумевает взаимодействие двух объектов: источника и приемника. Объект-приемник создает привязку к определенному свойству объекта-источника.



ежимы привязки

Свойство Mode объекта Binding, которое представляет режим привязки, может принимать следующие значения:

OneWay: свойство объекта-приемника изменяется после модификации свойства объекта-источника.

OneTime: свойство объекта-приемника устанавливается по свойству объекта-источника только один раз. В дальнейшем изменения в источнике никак не влияют на объект-приемник.

TwoWay: оба объекта - применки и источник могут изменять привязанные свойства друг друга.

OneWayToSource: объект-приемник, в котором объявлена привязка, меняет объект-источник.

Default: по умолчанию (если меняется свойство TextBox.Text, то имеет значение TwoWay, в остальных случаях OneWay).

9) Объясните назначение интерфейса INotifyPropertyChanged

Интерфейс INotifyPropertyChanged используется для уведомления клиентов(обычно привязывая клиентов) об изменении значения свойства.