Вариант 1.

Создайте приложение «Магазин».

1. Добавьте экран загрузки с логотипом магазина.

2. В первой Activity создать три редактируемых текстовых поля (EditText) для ввода пользователем регистрационных данных (телефона, имени и фамилии), кнопку «Регистрация» для запуска второй Activity. При нажатии на нее выполнить явный вызов второй Activity с передачей данных о пользователе (телефон, имя и фамилия). Добавьте кнопку «Пропустить».

3. Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки «Регистрация» должно программно меняться на «Войти».

4. Во второй Activity создать список товаров доступных к покупке. Предусмотрите возможность просмотра цен. По нажатию на одну из моделей она добавляется в корзину. Добавить кнопку «Корзина», по нажатию на которую перейти на новую Activity.

5. На третьей Activity предоставить выбрать товаров для покупки. Для этого вывести список всех товаров, добавленных в корзину с указанием их цены и возможностью множественного выбора. По окончании выбора пользователь нажимает на кнопку «Купить» и переходит на страницу оформления заказа.

6. На четвертой Activity отображается список товаров к покупке и финальная сумма. Можно вернуться назад и изменить свой выбор.

7. Есть кнопка «Отправить заказ», по нажатию на которую формируется текст сообщения с данными заказа, в качестве адресата задается адрес «организации» (придумайте его сами или используйте имеющийся). Можно послать письмо или отправить СМС-сообщение по вашему выбору. Если регистрация была пропущена, то по нажатию на эту кнопку, пользователя отправляют на первый экран для ввода регистрационных данных. (с помощью метода startActivityForResult).

8. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите отмену заказа и сброс выбранных пунктов в списках.

9. Создайте меню для управления переходами в приложении.

10. Главной страницей приложения должна быть вторая Activity, следовательно, она всегда должна быть в стеке, первая и четвертая храниться в стеке не должны. Все Activity в стеке должны быть в единственном экземпляре.

11. Замените иконку приложения и дайте ему нормальное название.

Вариант 2.

Создайте приложение «Покупки».

1. Добавьте экран загрузки с логотипом или картинкой.

2. В первую Activity добавить выпадающий список с выбором на какой день формируется список покупок, например «Сегодня, Завтра, На этой неделе». Добавьте кнопку для редактирования списка покупок

3. Во второй Activity выводится список покупок по выбранному ранее пункту с возможностью множественного выбора. Предусмотрите возможность добавления новых продуктов. Добавьте кнопку «Готово» по нажатию на которую перейти на новую Activity.

4. На третьей Activity отображается список продуктов, которые не были куплены. Можно вернуться назад и изменить свой выбор.

5. Добавить кнопку «Отправить отчет», по нажатию на которую формируется текст сообщения с данными о продуктах, которые еще нужно купить, в качестве адресата задается адрес «мамы» (придумайте его сами или используйте имеющийся). Можно послать письмо или отправить СМС-сообщение по вашему выбору. Можно добавить кнопку «Позвонить» по нажатию на которую откроется программа звонков с введенным номером «мамы».

6. Добавить кнопку «Перенести» и выпадающий список с вариантами, на какой срок можно перенести несделанные покупки, соответствующий выбору на первой странице. При этом пункт «сегодня» очищается и на него переносятся продукты записанные на завтра.

7. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите сброс выбранных пунктов в списках.

8. Реализовать сохранение данных в списках с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения.

9. Создайте меню для управления переходами в приложении.

10. Главной страницей приложения должна быть первая Activity, следовательно, она всегда должна быть в стеке, третья храниться в стеке не должна. Все Activity в стеке должны быть в единственном экземпляре.

11. Замените иконку приложения и дайте ему нормальное название.

Вариант 3.

Создайте приложение «Школа».

1. Добавьте экран загрузки с логотипом школы.

2. В первой Activity создать список дней (дата и день недели) на которые есть задания и пометки, а также редактируемое текстовое поле (EditText) для ввода пользователем новых дней для создания на них пометок. По нажатию на один из пунктов в списке перейти на новую Activity.

3. Во второй Activity создать список с различными заданиями или пометками (что принести, что сделать). Также добавьте редактируемое текстовое поле для ввода пользователем новых пометок. Список должен быть с возможностью множественного выбора.

4. Добавьте кнопку «Готово» по которой данные сохраняются и программа переходит на первую Activity, кнопку «Перенести на…» и выпадающий список с вариантами, на какой срок можно перенести невыполненные задания, и кнопку «Выполнено», по нажатию на которую список очищается и день удаляется из списка. При этом программа переходит на первую страницу.

5. При длительном нажатии на пункт списка выпадает контекстное меню для удаления или редактирования текущего пункта. Для редактирования открывается новое Activity (с помощью метода startActivityForResult).

6. В любой момент можно вернуться назад и изменить свой выбор.

7. Добавьте возможность «Позвонить учительнице» при выборе которого откроется программа звонков с введенным номером.

8. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите отмену заказа и сброс выбранных пунктов в списках.

9. Реализовать сохранение данных в списках с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения.

10. Создайте меню для управления переходами в приложении.

11. Главной страницей приложения должна быть вторая Activity, следовательно, она всегда должна быть в стеке, первая и четвертая храниться в стеке не должны. Все Activity в стеке должны быть в единственном экземпляре.

12. Замените иконку приложения и дайте ему нормальное название.

Вариант 4.

Создайте приложение «Список дел».

1. Добавьте экран загрузки с логотипом или картинкой.

2. В первую Activity добавить список с выбором на какой день (сегодня, завтра, на этой неделе) или в каком направлении (учеба, дом, подарки) формируется список дел. Также добавьте редактируемое текстовое поле (EditText) для ввода пользователем новых дней или тем для создания на них пометок. По нажатию на один пунктов в списке перейти на новую Activity.

3. Во второй Activity выводится список дел по выбранному ранее пункту с возможностью множественного выбора. Предусмотрите возможность добавления новых заданий.

4. Добавьте кнопку «Готово» по которой данные сохраняются и программа переходит на первую Activity, кнопку «Перенести на…» и выпадающий список с вариантами, на какой срок или в какую тему можно перенести невыполненные задания, и кнопку «Выполнено», по нажатию на которую список очищается и день удаляется из списка. При этом программа выдает сообщение «Кто Молодец? Я Молодец!» и переходит на первую страницу.

5. При длительном нажатии на пункт списка выпадает контекстное меню для удаления, редактирования текущего пункта или делегирования. Для редактирования открывается новое Activity (с помощью метода startActivityForResult). Для делегирования формируется текст сообщения с данными о делах, которые еще нужно сделать и открываются контакты для выбора того, кого бы вы хотели попросить их выполнить. Можно послать письмо или отправить СМС-сообщение по вашему выбору.

6. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите сброс выбранных пунктов в списках.

7. Реализовать сохранение данных в списках с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения.

8. Создайте меню для управления переходами в приложении.

9. Главной страницей приложения должна быть первая Activity, следовательно, она всегда должна быть в стеке, третья храниться в стеке не должна. Все Activity в стеке должны быть в единственном экземпляре.

10. Замените иконку приложения и дайте ему нормальное название.

Вариант 6.

Создайте приложение «Покупка автомобиля».

1. Добавьте экран загрузки с логотипом компании.

2. В первой Activity создать три редактируемых текстовых поля (EditText) для ввода пользователем регистрационных данных (телефона, имени и фамилии), кнопку «Регистрация» для запуска второй Activity. При нажатии на нее выполнить явный вызов второй Activity с передачей данных о пользователе (телефон, имя и фамилия). Добавьте кнопку «Пропустить».

Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки «Регистрация» должно программно меняться на «Войти».

3. Во второй активити создать список автомобилей доступных к покупке. Предусмотрите возможность просмотра цен на разные модели. По нажатию на одну из моделей в списке перейти на новую Activity.

4. На третьей Activity предоставить выбрать комплектацию автомобиля. Для этого вывести список возможных дополнительных опций с указанием их цены и возможностью множественного выбора. По окончании выбора пользователь нажимает на кнопку «Далее» и переходит на страницу оформления заказа.

5. На четвертой Activity отображается модель, состав комплектации и финальная сумма автомобиля. Можно вернуться назад и изменить свой выбор.

6. Есть кнопка «Отправить заказ», по нажатию на которую формируется текст сообщения с данными заказа, в качестве адресата задается адрес «организации» (придумайте его сами или используйте имеющийся). Можно послать письмо или отправить СМС-сообщение по вашему выбору. Если регистрация была пропущена, то по нажатию на эту кнопку, пользователя отправляют на первый экран для ввода регистрационных данных. (с помощью метода startActivityForResult).

7. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите отмену заказа и сброс выбранных пунктов в списках.

8. Создайте меню для управления переходами в приложении.

9. Главной страницей приложения должна быть вторая Activity, следовательно, она всегда должна быть в стеке, первая и четвертая храниться в стеке не должны. Все Activity в стеке должны быть в единственном экземпляре.

10. Замените иконку приложения и дайте ему нормальное название.

Вариант 7.

Создайте приложение «».

1. Добавьте экран загрузки с логотипом компании.

2. В первой Activity создать три редактируемых текстовых поля (EditText) для ввода пользователем регистрационных данных (телефона, имени и фамилии), кнопку «Регистрация» для запуска второй Activity. При нажатии на нее выполнить явный вызов второй Activity с передачей данных о пользователе (телефон, имя и фамилия). Добавьте кнопку «Пропустить».

Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки «Регистрация» должно программно меняться на «Войти».

3. Во второй активити создать список автомобилей доступных к покупке. Предусмотрите возможность просмотра цен на разные модели. По нажатию на одну из моделей в списке перейти на новую Activity.

4. На третьей Activity предоставить выбрать комплектацию автомобиля. Для этого вывести список возможных дополнительных опций с указанием их цены и возможностью множественного выбора. По окончании выбора пользователь нажимает на кнопку «Далее» и переходит на страницу оформления заказа.

5. На четвертой Activity отображается модель, состав комплектации и финальная сумма автомобиля. Можно вернуться назад и изменить свой выбор.

6. Есть кнопка «Отправить заказ», по нажатию на которую формируется текст сообщения с данными заказа, в качестве адресата задается адрес «организации» (придумайте его сами или используйте имеющийся). Можно послать письмо или отправить СМС-сообщение по вашему выбору. Если регистрация была пропущена, то по нажатию на эту кнопку, пользователя отправляют на первый экран для ввода регистрационных данных. (с помощью метода startActivityForResult).

7. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите отмену заказа и сброс выбранных пунктов в списках.

8. Создайте меню для управления переходами в приложении.

9. Главной страницей приложения должна быть вторая Activity, следовательно, она всегда должна быть в стеке, первая и четвертая храниться в стеке не должны. Все Activity в стеке должны быть в единственном экземпляре.

10. Замените иконку приложения и дайте ему нормальное название.

Оформление заказа Zayt$evDetail

1) Добавьте экран загрузки с логотипом компании.

2) Дальше появляется главный экран. Там можно посмотреть новости ZaytsevDetail, акции, красивые фотки машин.

3) При нажатии на кнопку "Услуги" открывается некст активити.

4) В активити создать список доступных услуг с наличием цен. При долгом нажатии на одну из услуг будет показана дополнительная информация о ней.

5) Пользователь с помощью чекбоксов выбирает интересующие его услуги (их может быть несколько) и нажимает кнопку "Записаться"

6) Во второй Activity создать 6 редактируемых текстовых поля (EditText) для ввода Фамилии, Имени, Отчества, номера телефона, марки автомобиля и его гос.номера.

Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки «Записаться» должно программно меняться на «Продолжить запись».

7) При вводе данных о клиенте и нажатии на кнопку "Далее" открывается треться активити, где будет вся информация о заявке (Имя,Фамилия,номер телефона, марка авто, гос номер, список выбранных услуг). Также должна высчитываться общая стоимость заказа. ИИИИИИИ, появляется календарик с выбором доступных дат.

8) После выбора даты и времени есть возможность нажать на чекбокс "отправить смс". После выбора этой функции и нажатии на кнопку "Отправить заявку" на телефон пользователя придет СМС со всеми данными, которые сформировались выше. Если чекбокс не выбирать а нажать просто "Отправить заявку" то смс никто не получит

9) Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите отмену заказа и сброс выбранных пунктов в списках.

10) Создайте меню для управления переходами в приложении.

11) Главной страницей приложения должна быть вторая Activity, следовательно, она всегда должна быть в стеке, первая и четвертая храниться в стеке не должны. Все Activity в стеке должны быть в единственном экземпляре.

12) Замените иконку приложения и дайте ему нормальное название.

Вариант 9.

Создайте приложение «Гостиница».

1. Добавьте экран загрузки с логотипом компании.

2. В первой Activity создать три редактируемых текстовых поля (EditText) для ввода пользователем регистрационных данных (телефона, имени и фамилии), кнопку «Регистрация» для запуска второй Activity. При нажатии на нее выполнить явный вызов второй Activity с передачей данных о пользователе (телефон, имя и фамилия). Добавьте кнопку «Пропустить».

3. Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки «Регистрация» должно программно меняться на «Войти».

4. Во второй Activity создать список номеров доступных к заказу. Предусмотрите возможность просмотра цен на разные варианты. По нажатию на один из номеров в списке перейти на новую Activity.

5. На третьей Activity вывести список возможных дополнительных опций с указанием их цены и возможностью множественного выбора. По окончании выбора пользователь нажимает на кнопку «Далее» и переходит на страницу оформления заказа.

6. На четвертой Activity отображается выбранный номер, состав комплектации и финальная сумма заказа. Можно вернуться назад и изменить свой выбор.

7. Есть кнопка «Отправить заказ», по нажатию на которую формируется текст сообщения с данными заказа, в качестве адресата задается адрес «отеля» (придумайте его сами или используйте имеющийся). Можно послать письмо или отправить СМС-сообщение по вашему выбору. Если регистрация была пропущена, то по нажатию на эту кнопку, пользователя отправляют на первый экран для ввода регистрационных данных. (с помощью метода startActivityForResult).

8. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите отмену заказа и сброс выбранных пунктов в списках.

9. Создайте меню для управления переходами в приложении.

10. Главной страницей приложения должна быть вторая Activity, следовательно, она всегда должна быть в стеке, первая и четвертая храниться в стеке не должны. Все Activity в стеке должны быть в единственном экземпляре.

11. Замените иконку приложения и дайте ему нормальное название.

Вариант 10.

Создайте приложение «Кинотеатр».

1. Добавьте экран загрузки с логотипом компании.

2. В первой Activity создать три редактируемых текстовых поля (EditText) для ввода пользователем регистрационных данных (телефона, имени и фамилии), кнопку «Регистрация» для запуска второй Activity. При нажатии на нее выполнить явный вызов второй Activity с передачей данных о пользователе (телефон, имя и фамилия). Добавьте кнопку «Пропустить».

3. Реализовать сохранение регистрационных данных пользователя в исходном Activity с помощью класса SharedPreferences и восстанавливать эту информацию при повторных запусках приложения. При этом название кнопки «Регистрация» должно программно меняться на «Войти».

4. Во второй Activity создать список фильмов для покупки билетов. Предусмотрите возможность просмотра цен на разные фильмы. По нажатию на один из фильмов в списке перейти на новую Activity.

5. На третьей Activity вывести список возможных дополнительных опций с указанием их цены и возможностью множественного выбора. По окончании выбора пользователь нажимает на кнопку «Далее» и переходит на страницу оформления заказа.

6. На четвертой Activity отображается выбранный фильм, дополнительные услуги и финальная сумма заказа. Можно вернуться назад и изменить свой выбор.

7. Есть кнопка «Отправить заказ», по нажатию на которую формируется текст сообщения с данными заказа, в качестве адресата задается адрес «отеля» (придумайте его сами или используйте имеющийся). Можно послать письмо или отправить СМС-сообщение по вашему выбору. Если регистрация была пропущена, то по нажатию на эту кнопку, пользователя отправляют на первый экран для ввода регистрационных данных. (с помощью метода startActivityForResult).

8. Предусмотрите сохранение данных при переходах с одной активити на другую и при поворотах экрана. Сделайте layout-ы для горизонтальной раскладки, если это необходимо. Предусмотрите отмену заказа и сброс выбранных пунктов в списках.

9. Главной страницей приложения должна быть вторая Activity, следовательно, она всегда должна быть в стеке, первая и четвертая храниться в стеке не должны. Все Activity в стеке должны быть в единственном экземпляре.

10. Замените иконку приложения и дайте ему нормальное название.