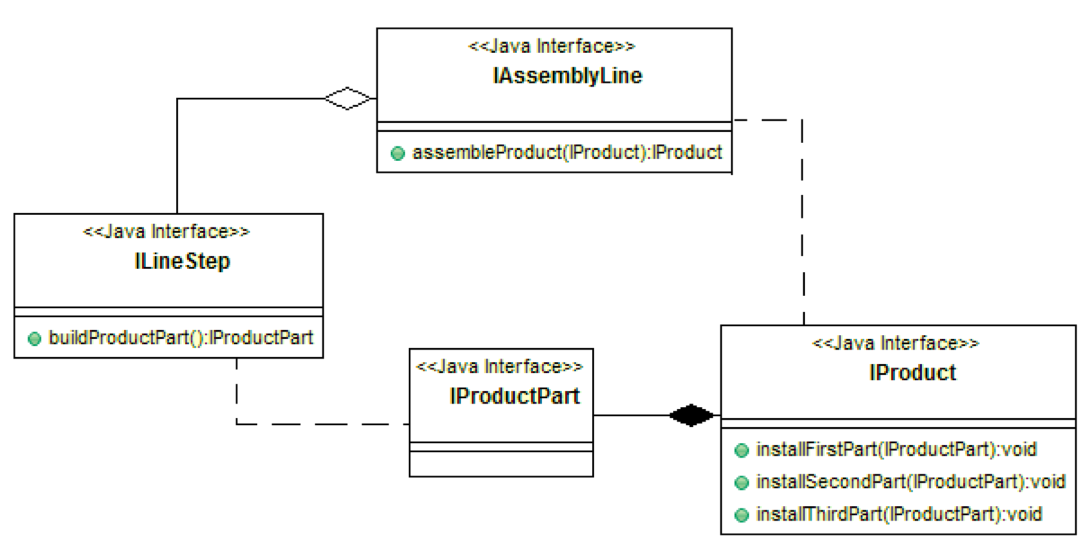
**Домашнее задание №19**

Задания выполняются в ветке **hw19**. Также необходимо создать pull request для ветки **hw19** в ветку **master (main)**.

Классы с решением заданий должны быть расположены в пакете **by.itacademy.hw19.**

Каждое задание выполняется в отдельном пакете, имеющем название **taskN**, где N - порядковый номер выполняемой задачи. При выполнение использовать принципы ООП.

1. Реализовать сборочную линию объектов, которая реализует интерфейсы, приведенные на следующей диаграмме:



Задача сборочной линии - собирать сложные продукты, состоящие из 3-х составляющих. Для установки каждой части использоваться специализированный класс (реализующий ILineStep) который поставляет необходимую часть через метод buildProductPart(). Для сборки продукта, его заготовка передается в метод assembleProduct(). Внутри метода, линия получает 3 необходимых составляющих и «устанавливает» их в соответствующие методы полученного продукта. После завершения «сборки», готовый продукт возвращается из метода assembleProduct().

1. Варианты сборочных линий:

1) Автомобили (кузов, шасси, двигатель)

2) Танки (корпус, двигатель, башня)

3) Ноутбуки (корпус, материнская плата, монитор)

4) Автоматические шариковые ручки (корпус, пружина, стержень)

5) Очки (корпус, линзы, дужки)

Реализовать отдельный поверочный класс с методом main() который сконструирует сборочную линию со всеми ее шагами, и протестирует ее запуском одного продукта на сборку. В конструктор сборочной линии должны передаваться 3 ссылки на необходимые сборочные шаги.

Весь процесс сборки должен детально выводиться пользователю на консоль.

2. Выбрать одно из ниже перечисленных заданий (или выбрать свою тему) и составить для него диаграммы прецедентов и классов.

\*\* Выбранная тема будет дорабатываться в последующих домашних заданиях.

1. **Электронный администратор гостиницы.**

Программа должна предоставлять возможность:

* Поселить в номер
* Выселить из номера
* Изменить статус номера на ремонтируемый/обслуживаемый
* Изменить цену номера или услуги
* Добавить номер или услугу

1. **Электронный книжный магазин.**

Программа должна предоставлять возможность:

* Списать книгу со склада (перевести в статус “отсутствует”)
* Создать заказ
* Отменить заказ
* Изменить статус заказа (новый, выполнен, отменен)
* Добавить книгу на склад (закрывает все запросы книги и меняет ее статус на “в наличии”)
* Оставить запрос на книгу

Дополнение:

Все доступные книги заранее определены. Они быть “в наличии” или “отсутствует”. Под “Оставить запрос на книгу” подразумевается создание запроса на книгу, что сейчас “отсутствует”.

При создании заказа с книгой, которой нет в наличии - автоматически создается запрос на эту книгу. Заказ не может быть завершен, до тех пор, пока запрос книги не будет выполнен.

1. **Электронный администратор автосервиса**

Программа должна предоставлять возможность:

* Добавить/удалить мастера
* Добавить/удалить место в гараже
* Добавить/удалить/закрыть/отменить/ заказ
* Сместить время выполнения заказов (из-за задержек в выполнении текущего)