

# Constructing Testable Objects

---



**Matthew Renze**

SOFTWARE CONSULTANT

@matthewrenze [www.matthewrenze.com](http://www.matthewrenze.com)



# Overview



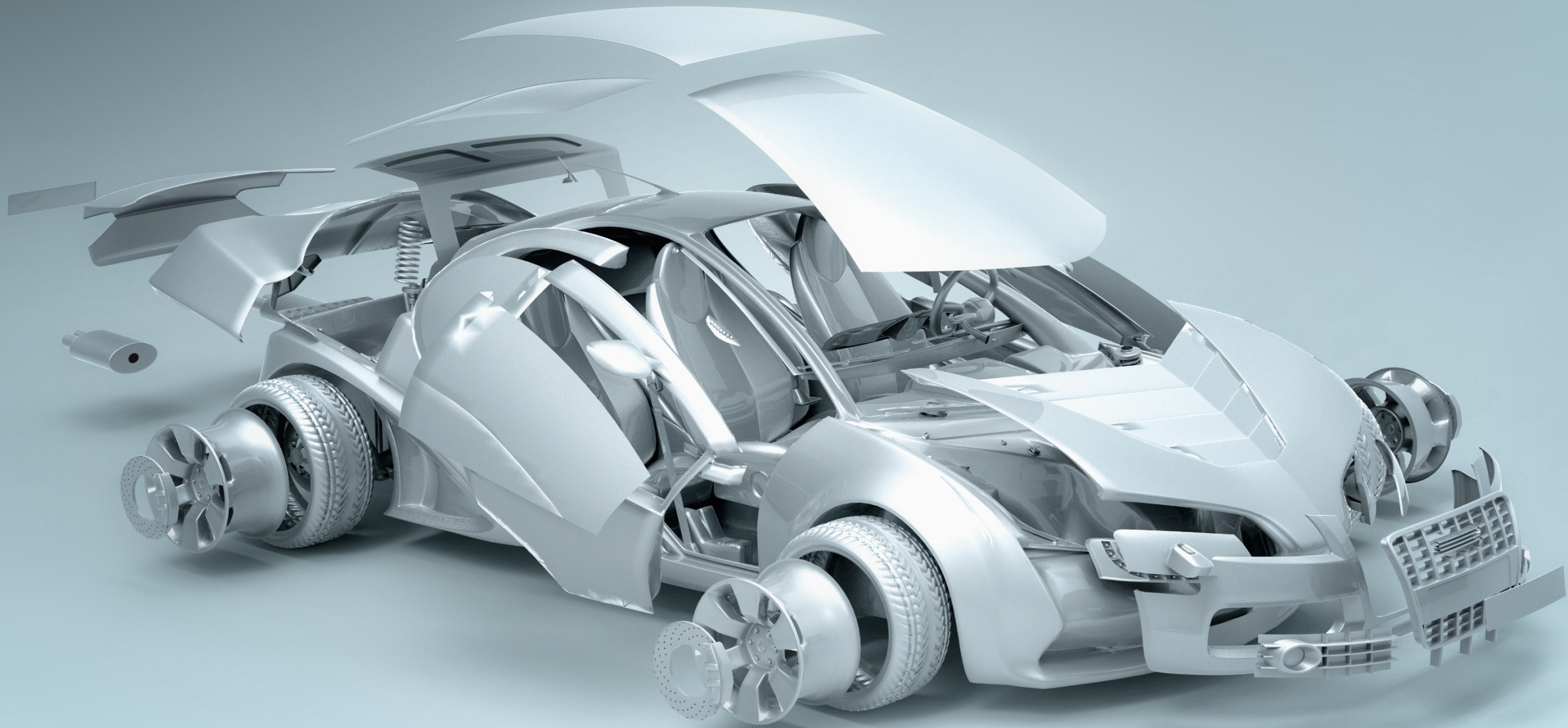
**Constructors**

**Problems**

**Symptoms**

**Solution**





# Constructors

Used to build objects

Prepares object for use



# Problems

**Creates tight coupling**

**Logic is difficult to test**

**Logic is difficult to set up**



# Symptoms

Keyword “new”

Logic in constructor

Any non-assignment code



# Solution

Inject dependencies

Avoid logic in constructor

Use factory, builder or IoC/DI

Don't mix construction and logic

Separate injectables vs. newables



# Injectable vs. Newable

**Injectable can ask for injectable**  
**... but should not ask for newable**  
**Newable can ask for newable**  
**... but should not ask for injectable**





## Invoice

Invoice #: 12345

Date: 01/01/2001

Customer: John Doe

Parts: \$100.00

Service: \$200.00

Discount: -\$50.00

**Total: \$250.00**

Terms: Net 30 days

Printed: 02/03/2001

*Overdue: Please pay ASAP!*

Print



## Invoice

Invoice #: 12345

Date: 01/01/2001

Customer: John Doe

Parts: \$100.00

Service: \$200.00

Discount: -\$50.00

**Total: \$250.00**

Terms: Net 30 days

Printed: 02/03/2001

*Overdue: Please pay ASAP!*

Print



# Show Print Invoice Command Class (Hard)



# Show Inventory Writer Class (Hard)



# Show Invoice Writer (Easy)



# Show Print Invoice Command (Easy)



# Show Invoice Writer Tests



Show Program 1 (Poor Man's DI)





# Show Program 2 (Factory)



Show Program 3 (DI/IoC)



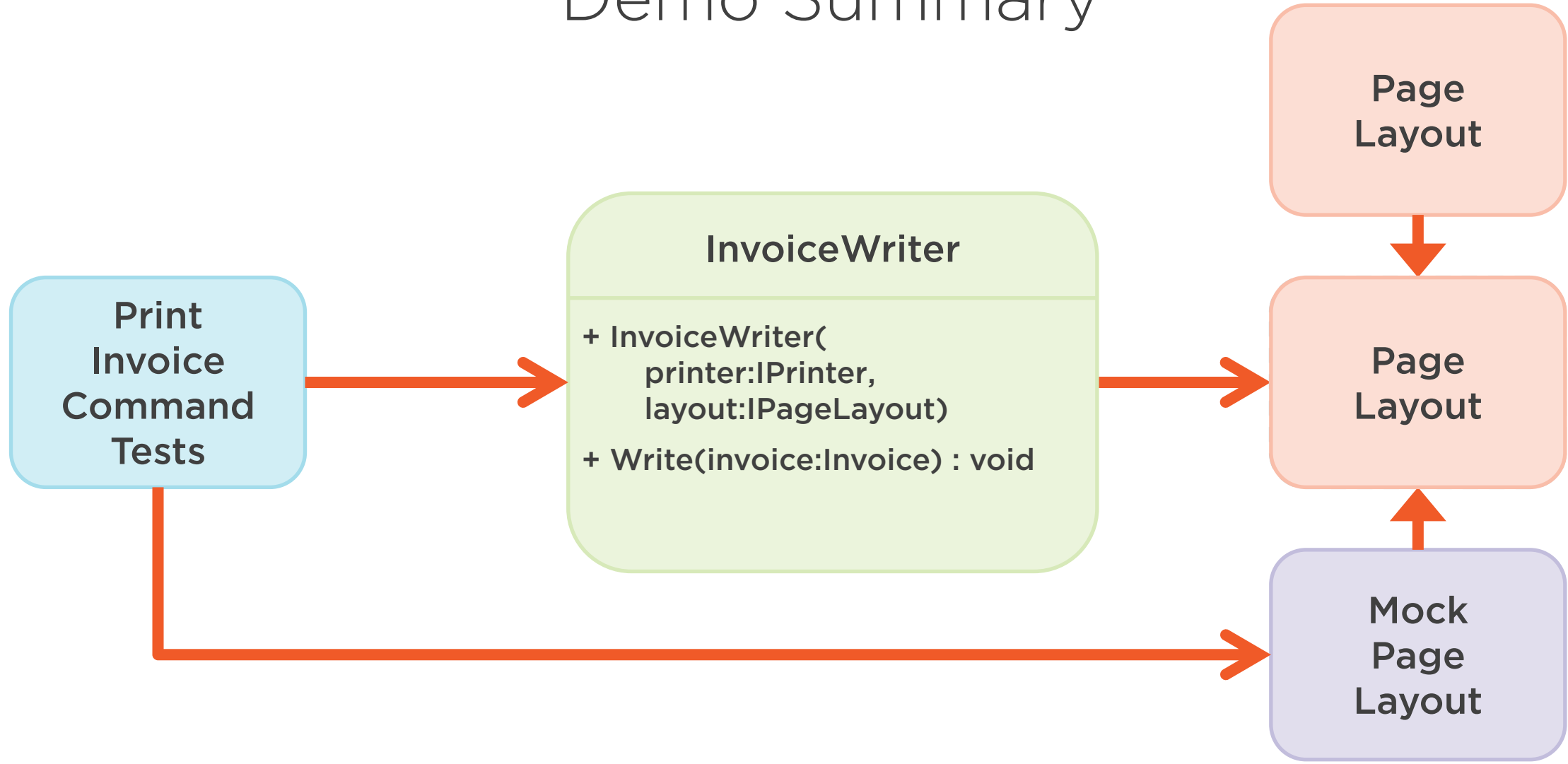
Show Program 4 (DI/loC with Conventions)



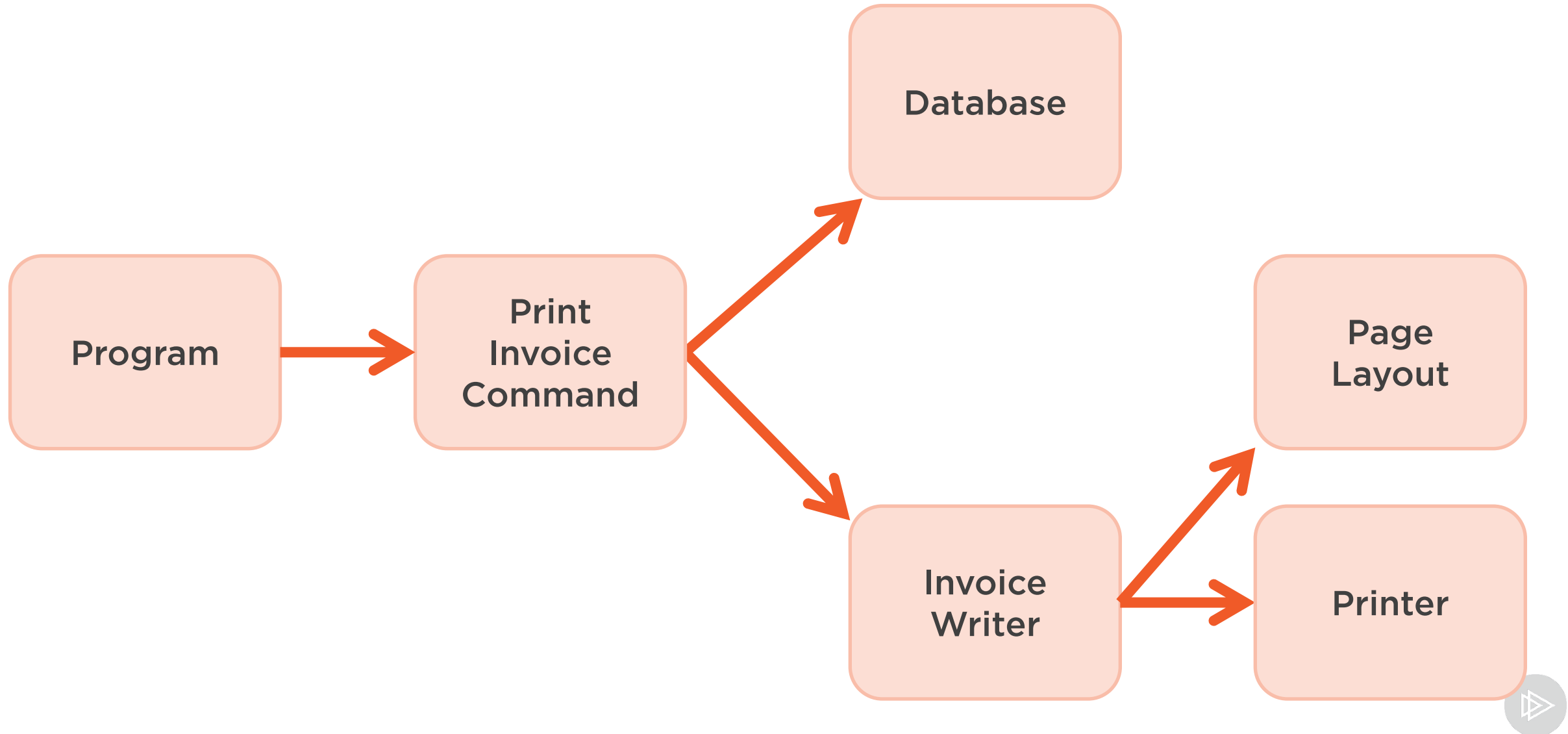
# Demo Summary



# Demo Summary



# Demo Summary



# Summary



**Simplify construction**

**Inject dependencies**

**Avoid logic in constructor**

**Don't mix construction and logic**

**Separate injectables vs. newables**

