**Arbeitsblatt**

**Java Design Patterns**

## Piazza 1

* Builder
  1. Pizza Hawaii hat einen krossen Teig, mit einer milden Sauce und als Belag Schinken mit Ananas
  2. Die scharfe Pizza besitzt einen dünnen Teig, eine scharfe Sauce und Peperoni und Salami als Belag.
* Composite
  1. Menu und MenuItem sind MenuComponents

## Piazza 2

* Facade
  1. Initialisiere alle Geräte
  2. Zum Fernsehschauen benötigen wird Popcorn. Schalte die Maschine ein und mache Popcorn. Dimme das Licht auf die Stufe 10 und die Leinwand ist unten. Schalte den Beamer ein und stelle ihn auf Breitbild. Der Verstärker muss eingeschaltet und DVD ausgewählt sein. Zum optimalen Sound benötigen wir Surround Sound und eine Lautstärke auf Stufe 5. Schalte letztendlich den DVD Player ein und spiele den Film ab.
  3. Zum Beenden des Filmes schalte die Popcorn Maschine, den Beamer, den Verstärker aus und schalte das Licht ein. Die Leinwand muss nach oben und die DVD muss herausgeholt werden. Schalte den DVD Player aus.
  4. Zum Musik hören schalte das Licht und den Verstärker ein. Stelle den Modus CD ein und stelle die Lautstärke auf Stufe 5. Setze den Surround Sound, schalte den CD Player ein und spiele die CD ab.
  5. Zum Beenden schalte den Verstärker aus, hole die CD heraus und schalte den CD Player aus.
  6. Zum Radio hören schalte den Tuner und den Verstärker ein und wähle die Frequenz aus. Reguliere die Lautstärke auf 5 und stelle den Verstärker auf den Modus Tuner.
  7. Zum Beenden des Radios schalte den Verstärker und den Tuner aus.
* Obeserver
  1. Die Statistik berechnet sich wie folgt. Summiere die Temperatur mit der tempSum und erhöhe die numReadings um 1. Ist die gegebene Temperatur höher als maxTemp, setze maxTemp auf die gegebene Temperatur. Ist die gegebene Temperatur kleiner als die minTemp, setze minTemp auf die gegebene Temperatur. Rufe display auf.
  2. Der HeatIndex berechnet sich wie folgt: Berechne den heatIndex mit Hilfe von computeHeatIndex und rufe display auf.
  3. Der Forecast berechnet sich wie folgt: Setze lastPressure auf currentPressure und currentPressure auf den gegebenen Druck und rufe display auf.
  4. Die Currentcondition berechnet sich wie folgt: Setze die gegebene Temperatur und die Luftfeuchte und rufe display auf.

## Piazza 3

* Factory
  1. SalaryFactory soll das Gehalt bezogen auf den Titel zurückliefern
  2. Optional: Anstatt eines einfachen Integers erstelle ein Interface Salary und erzeuge darauf konkrete Implementierungen, die das jeweilige Gehalt als Objekt zurückliefern
  3. Tipp: Durch Komposition erhältst du mehr Flexibilität in der Employee Klasse
  4. Optional: Verwende die Java 8 Supplier, um aus der Factory Objekte zu erzeugen
* Strategy
  1. Das Duck Interface soll durch eine abstrakte Klasse ersetzt werden
  2. Anstatt den einzelnen Duck Implementierungen durch die beiden Interfaces (FlyBehavior und QuackBehavior) konkrete Eigenschaften mitzugeben, sollen sie mehr Flexibilität durch Komposition erhalten, damit erst zur Laufzeit das konkrete Verhalten ermittelt wird
  3. Setze die konkreten Verhaltensweisen bei der Initialisierung der Ducks fest
  4. Optional: Nutze Lambdas anstatt konkrete Implementierung, um die Strategy zu setzen

## Optional

* Builder
  1. Die HttpOptions können mit der Fluent Schreibweise um Konfigurationen erweitert werden
  2. Host und Port sind Pflichtattribute
* Visitor
  1. Implementiere alle CarElements
  2. Implementiere ein Auto bestehend aus 4 Rädern, einem Body und einem Motor
  3. Iteriere über alle CarElements und akzeptiere den Visitor
* Adapter
  1. Adaptiere das Verhalten einer Ente auf einen Truthahn
  2. Adaptiere das Verhalten eines Truthahns auf eine Ente
  3. Adaptiere einen String zu Boolean mit Hilfe von Lambdas