**/Thoughtworks - Vapasi Accelerator Trainer**

**SOLID**

* [The SOLID Principles](http://www.blackwasp.co.uk/SOLIDPrinciples.aspx)

[**TDD**](https://docs.google.com/document/d/13y3LZo7P2nzUTANBp2SC-yg3IS03w60dByIobarZZkc/edit)

* [Testing strategy](https://medium.com/@netxm/how-to-get-started-with-software-testing-9fa1ce4f2a64)
* [Test Driven Development](https://medium.com/hackernoon/introduction-to-test-driven-development-tdd-61a13bc92d92)
* [Prepare for testing | IntelliJ IDEA](https://www.jetbrains.com/help/idea/testing.html#add-testing-libraries)
* [IntelliJ IDEA. Writing Tests with JUnit 5](https://www.youtube.com/watch?v=we3zJE3hlWE)

**OOP**

* [Object Oriented Design](https://www.geeksforgeeks.org/oops-object-oriented-design/)

**Pair Programming**

* [Effective Navigation in Pair Programming](https://www.thoughtworks.com/insights/blog/effective-navigation-in-pair-programming)
* [Remote Pairing 101](https://www.thoughtworks.com/insights/blog/remote-pairing-101)
* [7 Principles](https://www.thoughtworks.com/insights/blog/seven-principles-pair-programming-etiquette)

**Naming** **Conventions**

* [7 Popular Strategies: Unit Test Naming Conventions - DZone Agile](https://dzone.com/articles/7-popular-unit-test-naming)

[**Shortcuts**](https://docs.google.com/spreadsheets/u/0/d/1HbOeulnxcEx_zuqdD9oC8ES1qKuHgcgDgBV0DlhAfTA/edit)

* [**Intellij\_basic\_shortcuts\_prime\_number\_brinda\_yash.mp4**](https://drive.google.com/file/d/1naRKCKil_L39O_cjtq5YRj7oPC2erk0Y/view?usp=sharing)
* [Zoom Shortcuts](https://docs.google.com/document/d/1APuE7J3raNy2q2RQwCOz2W4SV5rLcXZH_N-kdPjOJBM/edit#heading=h.hcchn0tkwdzd)
* [**Vim cheatsheet**](https://devhints.io/vim)

**Gradle**

* [**gradleProjectCreation\_Yash.mp4**](https://drive.google.com/file/d/1bxr7emU9C3RorZkvPv510cmEixlRAeyw/view?usp=sharing)

**Processing**

* [5 - What is Processing.mp4](https://drive.google.com/file/d/1Pjmzbn_bHFsTohcAoq2qVFkmM-eFMT8x/view?usp=sharing)
* [6 - installing processing.mp4](https://drive.google.com/file/d/1e6ik2s-NvEJP8s_Vqd-yx1ffzqtjfOPY/view?usp=sharing)
* [7 - processing programming model.mp4](https://drive.google.com/file/d/1zKR73YxRZbzAgRwf8hJ6AVIQVAK6qusu/view?usp=sharing)
* [8 - challenge for you.mp4](https://drive.google.com/file/d/1SH9Y4QuC4OH5U94w2mlOaeJZMqMgssJV/view?usp=sharing)

[**GIT**](https://docs.google.com/document/d/1sl-f22B6ZlWdebTC58Vpj6z1l4Cmy3V_n5zMS-TABY0/edit)

* [Working with Protected Branches](https://www.nih-cfde.org/resource/working-with-protected-branches/)

**Command Line**

* h[ttps://explainshell.com](https://explainshell.com)
* [16.4. Text Processing Commands](https://tldp.org/LDP/abs/html/textproc.html)

**Design Rule**s

* [Domain Driven Design and Domain Modeling](https://dddcommunity.org/)
* [chcordova/design-patterns-head-first-book - Java 8 Examples](https://github.com/chcordova/design-patterns-head-first-book)
* [Design Pattern Interview Questions - YouTube](https://youtube.com/playlist?list=PLyHJZXNdCXselvpjEhygQ9GlDQihtQj6i)
* [Abstract Factory Design Pattern](https://www.youtube.com/watch?v=xbjAsdAK4xQ)
* [Factory Design Pattern](https://www.youtube.com/watch?v=ub0DXaeV6hA)
* [Yagni](https://martinfowler.com/bliki/Yagni.html)
* [BeckDesignRules](https://martinfowler.com/bliki/BeckDesignRules.html)
* [HeadFirstDesignPatterns.pdf](http://users/parulbhargava/Downloads/HeadFirstDesignPatterns.pdf)

**Learn Programming Languages**

* [Java-Books/Head First Java.pdf at master](https://github.com/Nitin96Bisht/Java-Books/blob/master/Head%20First%20Java.pdf)
* [Spring Framework - YouTube](https://youtube.com/playlist?list=PLC97BDEFDCDD169D7)
* <https://learnpythonthehardway.org/book/appendixa.html>