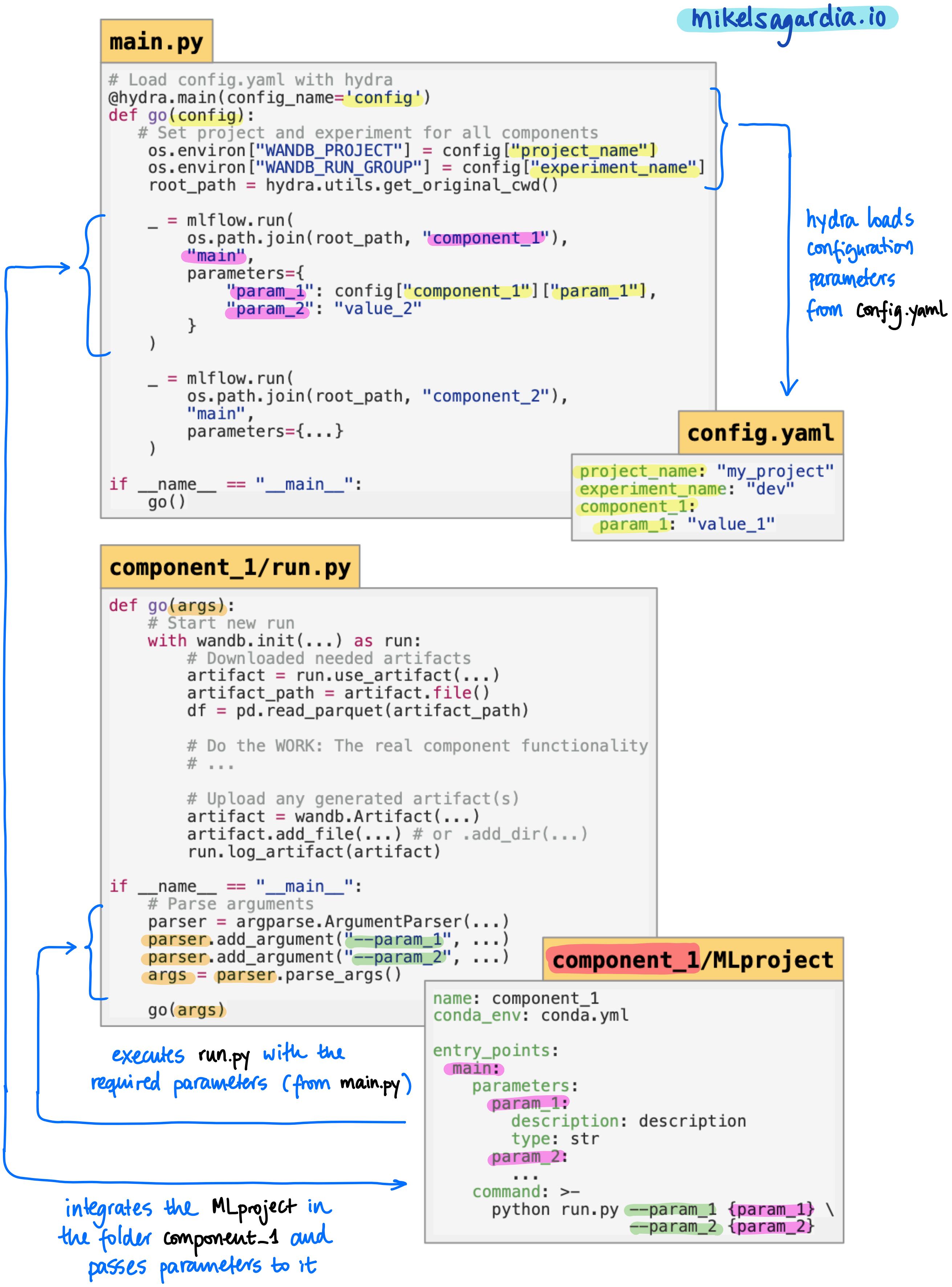

main.py

```python
# Load config.yaml with hydra
@hydra.main(config_name='config')
def go(config):
    # Set project and experiment for all components
    os.environ["WANDB_PROJECT"] = config["project_name"]
    os.environ["WANDB_RUN_GROUP"] = config["experiment_name"]
    root_path = hydra.utils.get_original_cwd()

    _ = mlflow.run(
        os.path.join(root_path, "component_1"),
        "main",
        parameters={
            "param_1": config["component_1"]["param_1"],
            "param_2": "value_2"
        }
    )

    _ = mlflow.run(
        os.path.join(root_path, "component_2"),
        "main",
        parameters={...}
    )

if __name__ == "__main__":
    go()
```

hydra loads configuration parameters from config.yaml

config.yaml

```yaml
project_name: "my_project"
experiment_name: "dev"
component_1:
  param_1: "value_1"
```

component_1/run.py

```python
def go(args):
    # Start new run
    with wandb.init(...) as run:
        # Downloaded needed artifacts
        artifact = run.use_artifact(...)
        artifact_path = artifact.file()
        df = pd.read_parquet(artifact_path)

        # Do the WORK: The real component functionality
        # ...

        # Upload any generated artifact(s)
        artifact = wandb.Artifact(...)
        artifact.add_file(...) # or .add_dir(...)
        run.log_artifact(artifact)

if __name__ == "__main__":
    # Parse arguments
    parser = argparse.ArgumentParser(...)
    parser.add_argument("--param_1", ...)
    parser.add_argument("--param_2", ...)
    args = parser.parse_args()

    go(args)
```

component_1/MLproject

```yaml
name: component_1
conda_env: conda.yml

entry_points:
  main:
    parameters:
      param_1:
        description: description
        type: str
      param_2:
        ...
    command: >-
      python run.py --param_1 {param_1} \
                    --param_2 {param_2}
```

executes run.py with the required parameters (from main.py)

integrates the MLproject in the folder component_1 and passes parameters to it