

# Test Time tool chain evolve

Task: Lowest LV bag price by country

```
def search_product(query):
    candidates = web_search(query=query)

    best = candidates[0]
    return {
        "product_id": best["id"],
        "name": best["title"],
        "brand": best.get("brand"),
    }
```

```
def select_region_site(product: dict,
country: str) -> str:
    if "product_id" not in product:
        return ""
    return resolve_product_url(
        product_id=product["product_id"],
        region=country
    )
```

```
def normalize_price(raw,
target_currency):
    price, cur = raw.get("price"),
raw.get("currency")
    if cur == target_currency:
        return {"price": price,
"currency": cur, "confidence":
raw.get("confidence", 1.0)}
    rate = foreign_exchange(source=cur,
target=target_currency)
    return {
        "price": round(price * rate, 2),
        "currency": target_currency,
        "confidence":
raw.get("confidence", 1.0),
    }
```

```
def query_price(url: str) -> dict:
    if not url:
        return {"price": None,
"currency": None, "confidence": 0.0}

    price_info = fetch_price(url)
    return {
        "price": price_info["price"],
        "currency":
price_info["currency"],
        "confidence":
price_info.get("confidence", 1.0),
    }
```







# Iterative Macro Composition

 bag\_price\_checker

```
def bag_price_checker(country) -> dict:
    product = search_product("bag")
    localized = select_region_site(product, country)
    raw = query_price(localized)
    norm = normalize_price(raw,
target_currency="USD")

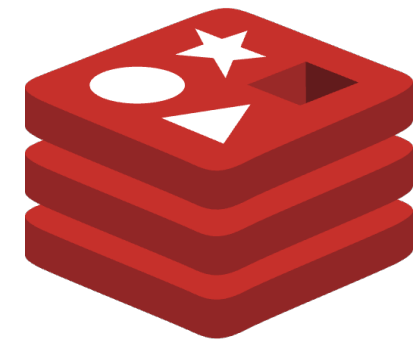
    return {
        "country": country,
        "price": norm["price"],
        "currency": norm["currency"],
        "confidence": norm["confidence"],
    }
```

# Macro Library

-  gender\_balance\_checker
-  geo\_demographic\_mapper
-  bag\_price\_checker
-  name\_quick\_profile
-  demographic\_car\_builder
-  multi-name\_batch-stats
- .....

Tool chain  
execution

Sample Task



Task Library

Execute

Return

If Fail  
Re-explore

If Pass  
Save new skill

Coding Env

Coding Verifier

Usable Macro retrieval