

```
class Campaign:
```

```
    def __init__(self, campaign_id, movie_title, budget, revenue, impressions, clicks):
```

```
        self.campaign_id = campaign_id
```

```
        self.movie_title = movie_title
```

```
        self.budget = budget
```

```
        self.revenue = revenue
```

```
        self.impressions = impressions
```

```
        self.clicks = clicks
```

```
class CampaignManager:
```

```
    def __init__(self):
```

```
        self.campaigns = {}
```

```
    def add_campaign(self, campaign):
```

```
        if campaign.campaign_id in self.campaigns:
```

```
            raise ValueError("Campaign ID already exists.")
```

```
        self.campaigns[campaign.campaign_id] = campaign
```

```
    def get_campaign(self, campaign_id):
```

```
        return self.campaigns.get(campaign_id, None)
```

```
    def update_campaign(self, campaign_id, **kwargs):
```

```
        campaign = self.get_campaign(campaign_id)
```

```
        if not campaign:
```

```
            raise ValueError("Campaign not found.")
```

```
        for key, value in kwargs.items():
```

```
            setattr(campaign, key, value)
```

```
    def delete_campaign(self, campaign_id):
```

```
        if campaign_id in self.campaigns:
```

```
            del self.campaigns[campaign_id]
```

```

def analyze_effectiveness_of_marketing(self, campaign_id):
    campaign = self.get_campaign(campaign_id)
    if not campaign:
        return None
    ROI = (campaign.revenue - campaign.budget) / campaign.budget if campaign.budget else 0
    CTR = campaign.clicks / campaign.impressions if campaign.impressions else 0
    return {"ROI": ROI, "CTR": CTR}

def optimize_marketing_strategies(self, strategy_data):
    # Placeholder for optimization logic
    return strategy_data

# Main code to demonstrate functionality
if __name__ == "__main__":
    # Create a CampaignManager instance
    manager = CampaignManager()

    # Add a new campaign
    campaign1 = Campaign(1, "Movie A", 100000, 300000, 10000, 500)
    manager.add_campaign(campaign1)

    # Retrieve and display the campaign
    retrieved_campaign = manager.get_campaign(1)
    print(f"Retrieved Campaign: {retrieved_campaign.movie_title}, Budget: {retrieved_campaign.budget}, Revenue: {retrieved_campaign.revenue}")

    # Update the campaign's revenue
    manager.update_campaign(1, revenue=500000)
    updated_campaign = manager.get_campaign(1)
    print(f"Updated Campaign Revenue: {updated_campaign.revenue}")

```

```
# Analyze the effectiveness of the marketing campaign

analysis = manager.analyze_effectiveness_of_marketing(1)

print(f"Marketing Effectiveness Analysis for Campaign 1: ROI = {analysis['ROI']:.2f}, CTR = {analysis['CTR']:.2f}")


# Delete the campaign

manager.delete_campaign(1)

print(f"Deleted Campaign 1, Now retrieved: {manager.get_campaign(1)}")
```