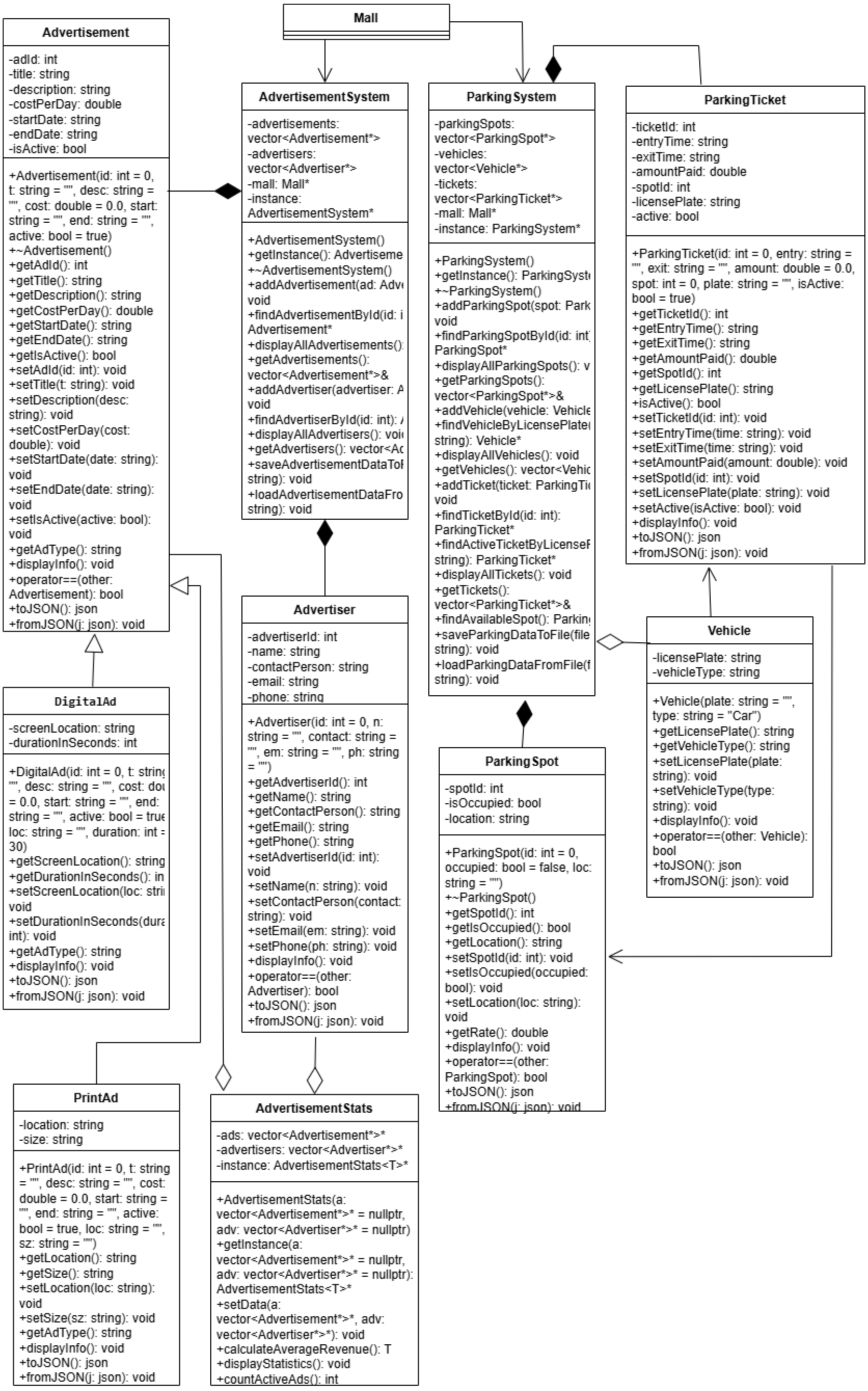
Mall
Advertisement
-adId: int
-title: string
-description: string
-costPerDay: double
-startDate: string
-endDate: string
-isActive: bool
+Advertisement(id: int = 0, t: string = "", desc: string = "", cost: double = 0.0, start: string = "", end: string = "", active: bool = true)
+~Advertisement()
+getAdId(): int
+getTitle(): string
+getDescription(): string
+getCostPerDay(): double
+getStartDate(): string
+getEndDate(): string
+getIsActive(): bool
+setAdId(id: int): void
+setTitle(t: string): void
+setDescription(desc: string): void
+setCostPerDay(cost: double): void
+setStartDate(date: string): void
+setEndDate(date: string): void
+setIsActive(active: bool): void
+getAdType(): string
+displayInfo(): void
+operator==(other: Advertisement): bool
+toJSON(): json
+fromJSON(j: json): void
AdvertisementSystem
-advertisements: vector<Advertisement*>
-advertisers: vector<Advertiser*>
-mall: Mall*
-instance: AdvertisementSystem*
+AdvertisementSystem()
+getInstance(): Advertiseme
+~AdvertisementSystem()
+addAdvertisement(ad: Adv
void
+findAdvertisementById(id: i
Advertisement*
+displayAllAdvertisements()
+getAdvertisements(): vector<Advertisement*>&
+addAdvertiser(advertiser: A
void
+findAdvertiserById(id: int): A
+displayAllAdvertisers(): voi
+getAdvertisers(): vector<Ac
+saveAdvertisementDataTo
string): void
+loadAdvertisementDataFro
string): void
ParkingSystem
-parkingSpots: vector<ParkingSpot*>
-vehicles: vector<Vehicle*>
-tickets: vector<ParkingTicket*>
-mall: Mall*
-instance: ParkingSystem*
+ParkingSystem()
+getInstance(): ParkingSyst
+~ParkingSystem()
+addParkingSpot(spot: Park
void
+findParkingSpotById(id: int
ParkingSpot*
+displayAllParkingSpots(): v
+getParkingSpots(): vector<ParkingSpot*>&
+addVehicle(vehicle: Vehicle
+findVehicleByLicensePlate(
string): Vehicle*
+displayAllVehicles(): void
+getVehicles(): vector<Vehic
+addTicket(ticket: ParkingTi
void
+findTicketById(id: int): ParkingTicket*
+findActiveTicketByLicenseF
string): ParkingTicket*
+displayAllTickets(): void
+getTickets(): vector<ParkingTicket*>&
+findAvailableSpot(): Parkin
+saveParkingDataToFile(file
string): void
+loadParkingDataFromFile(f
string): void
ParkingTicket
-ticketId: int
-entryTime: string
-exitTime: string
-amountPaid: double
-spotId: int
-licensePlate: string
-active: bool
+ParkingTicket(id: int = 0, entry: string = "", exit: string = "", amount: double = 0.0, spot: int = 0, plate: string = "", isActive: bool = true)
+getTicketId(): int
+getEntryTime(): string
+getExitTime(): string
+getAmountPaid(): double
+getSpotId(): int
+getLicensePlate(): string
+isActive(): bool
+setTicketId(id: int): void
+setEntryTime(time: string): void
+setExitTime(time: string): void
+setAmountPaid(amount: double): void
+setSpotId(id: int): void
+setLicensePlate(plate: string): void
+setActive(isActive: bool): void
+displayInfo(): void
+toJSON(): json
+fromJSON(j: json): void
Advertiser
-advertiserId: int
-name: string
-contactPerson: string
-email: string
-phone: string
+Advertiser(id: int = 0, n: string = "", contact: string = "", em: string = "", ph: string = "")
+getAdvertiserId(): int
+getName(): string
+getContactPerson(): string
+getEmail(): string
+getPhone(): string
+setAdvertiserId(id: int): void
+setName(n: string): void
+setContactPerson(contact: string): void
+setEmail(em: string): void
+setPhone(ph: string): void
+displayInfo(): void
+operator==(other: Advertiser): bool
+toJSON(): json
+fromJSON(j: json): void
Vehicle
-licensePlate: string
-vehicleType: string
+Vehicle(plate: string = "", type: string = "Car")
+getLicensePlate(): string
+getVehicleType(): string
+setLicensePlate(plate: string): void
+setVehicleType(type: string): void
+displayInfo(): void
+operator==(other: Vehicle): bool
+toJSON(): json
+fromJSON(j: json): void
DigitalAd
-screenLocation: string
-durationInSeconds: int
+DigitalAd(id: int = 0, t: string = "", desc: string = "", cost: dou = 0.0, start: string = "", end: string = "", active: bool = true loc: string = "", duration: int = 30)
+getScreenLocation(): string
+getDurationInSeconds(): in
+setScreenLocation(loc: stri void
+setDurationInSeconds(dura int): void
+getAdType(): string
+displayInfo(): void
+toJSON(): json
+fromJSON(j: json): void
Parking Spot
-spotId: int
-isOccupied: bool
-location: string
+ParkingSpot(id: int = 0, occupied: bool = false, loc: string = "")
+~ParkingSpot()
+getSpotId(): int
+getIsOccupied(): bool
+getLocation(): string
+setSpotId(id: int): void
+setIsOccupied(occupied: bool): void
+setLocation(loc: string): void
+getRate(): double
+displayInfo(): void
+operator==(other: ParkingSpot): bool
+toJSON(): json
+fromJSON(j: json): void
PrintAd
-location: string
-size: string
+PrintAd(id: int = 0, t: string = "", desc: string = "", cost: double = 0.0, start: string = "", end: string = "", active: bool = true, loc: string = "", sz: string = "")
+getLocation(): string
+getSize(): string
+setLocation(loc: string): void
+setSize(sz: string): void
+getAdType(): string
+displayInfo(): void
+toJSON(): json
+fromJSON(j: json): void
AdvertisementStats
-ads: vector<Advertisement*>*
-advertisers: vector<Advertiser*>*
-instance: AdvertisementStats<T>*
+AdvertisementStats(a: vector<Advertisement*>* = nullptr, adv: vector<Advertiser*>* = nullptr)
+getInstance(a: vector<Advertisement*>* = nullptr, adv: vector<Advertiser*>* = nullptr): AdvertisementStats<T>*
+setData(a: vector<Advertisement*>*, adv: vector<Advertiser*>*): void
+calculateAverageRevenue(): T
+displayStatistics(): void
+countActiveAds(): int