Awesome—here’s a compact, “Cursor‑ready” implementation brief you can paste into your repo as a /docs/odds-ingestion-spec.md (or into a Cursor task). It’s opinionated around **The Odds API v4** and your constraints:

* refresh **game odds every 5 minutes**
* include **player props** (esp. PrizePicks & DraftKings Pick6)
* keep **under ~100,000 monthly credits**
* also pull **historical** snapshots **every 2 hours**
* target **NFL, NBA, MLB** first

All API specifics (endpoints, costs, update intervals, bookmaker regions, market keys) below are pulled directly from the official docs and reference pages. Citations appear inline.

**Odds Ingestion — Implementation Brief (The Odds API v4)**

**0) Goals & constraints**

* **Leagues (phase 1):** NFL, NBA, MLB
* **Live refresh:**
  + **Game odds** (moneyline, spreads, totals) every **5 minutes**
  + **Player props**: budget‑aware cadence (below)
* **Historical snapshots:** every **2 hours**
* **Coverage breadth:** all US books; include DFS (PrizePicks, DraftKings Pick6)
* **Monthly budget cap:** ≤ **100,000** credits
* **Primary source:** The Odds API v4 [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**1) What we’ll ingest**

**A) Game odds (featured markets)**

* Markets: h2h (moneyline), spreads, totals
* Endpoint: **GET /v4/sports/{sport}/odds**
* Cost: **1 credit × (#markets specified) × (#regions)** (e.g., 3 markets × 2 regions = **6 credits**/call). Empty responses don’t cost. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* Update frequency guidance: featured markets update ~**60s pre‑match** and **~40s in‑play**; interval ramps down from ~6 hours before game. [The Odds API](https://the-odds-api.com/sports-odds-data/update-intervals.html?utm_source=chatgpt.com)

**B) Player props (additional markets, incl. DFS)**

* Endpoint: **GET /v4/sports/{sport}/events/{eventId}/odds**
* Cost: **[#unique markets returned] × (#regions)** per call (responses with empty data don’t cost). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* Additional markets are per‑event (not league‑wide) and update ~**60s** pre‑match & in‑play. [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html) [The Odds API](https://the-odds-api.com/sports-odds-data/update-intervals.html?utm_source=chatgpt.com)
* DFS coverage (region us\_dfs): **PrizePicks (prizepicks), DraftKings Pick6 (pick6), Underdog (underdog)**. [The Odds API](https://the-odds-api.com/sports-odds-data/bookmaker-apis.html)

**Common player‑prop market keys** (subset we’ll start with):

* **NFL:** player\_pass\_yds, player\_pass\_tds, player\_rush\_yds, player\_receptions, player\_reception\_yds, player\_anytime\_td (and more listed in docs). [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)
* **NBA:** player\_points, player\_rebounds, player\_assists, player\_threes, player\_points\_rebounds\_assists (+ alternates). [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)
* **MLB:** pitcher\_strikeouts, batter\_total\_bases, batter\_hits, batter\_home\_runs (+ alternates). [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)

**C) Historical odds (snapshots)**

* **Featured markets, league‑wide:**
  + **GET /v4/historical/sports/{sport}/odds**
  + Cost: **10 × (#markets specified) × (#regions)** per call; snapshots available at **5‑minute** intervals since Sep 2022 (10‑min prior to that). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/) [The Odds API](https://the-odds-api.com/historical-odds-data/?utm_source=chatgpt.com)
* **Per‑event historical (incl. props):**
  + **GET /v4/historical/sports/{sport}/events/{eventId}/odds** with date= snapshot (closest ≤ given timestamp).
  + Cost: **10 × (#unique markets returned) × (#regions)**. **Additional markets (player props, alternates)** available since **2023‑05‑03T05:30:00Z**; snapshots every **5 minutes**. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/?utm_source=chatgpt.com) [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* Discover event ids at a past timestamp via: **GET /v4/historical/sports/{sport}/events?date=...** (cost **1**; no odds). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**D) Scores (for grading game picks)**

* Endpoint: **GET /v4/sports/{sport}/scores**
* Cost: **1** credit; **2** if daysFrom (to include completed games back **1–3 days**) is set. **Live scores update ~30s**. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Important:** Scores are **team‑level**; **no player stats** in this API. For **player‑prop grading**, integrate a player‑stats source (e.g., SportsDataIO, Sportradar, etc.). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**2) Bookmakers & regions to request**

* For **US sportsbooks** use regions: us and us2 (adds more US books like ESPN BET, BetPARX, Hard Rock, etc.). [The Odds API](https://the-odds-api.com/sports-odds-data/bookmaker-apis.html)
* For **DFS** use region: us\_dfs (PrizePicks, DraftKings Pick6, Underdog). [The Odds API](https://the-odds-api.com/sports-odds-data/bookmaker-apis.html)
* You can also target exact bookmakers via bookmakers=; if you pass explicit lists, **every 10 bookmakers counts as 1 region** for cost purposes. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**3) Polling plan (budget‑aware)**

The plan below hits your **5‑minute** target for **game odds** while keeping **player props** & **history** inside ~100k credits/month. We also use free endpoints to decide when to poll.

**Cadence**

1. **Discovery (free):**
   * Every **2 minutes**, per sport: **GET /v4/sports/{sport}/events** to know which events are upcoming/live and their eventIds. (**Free**). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
2. **Game odds (featured markets):**
   * When at least one event exists today: every **5 minutes** per sport call  
     **/v4/sports/{sport}/odds?regions=us,us2&markets=h2h,spreads,totals**.  
     Cost per call: **3 markets × 2 regions = 6** credits. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
   * Use commenceTimeFrom/To to bound to today—doesn’t change cost but reduces payload. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
3. **Player props (additional markets, incl. DFS):**
   * For each **event** in **[T‑180m, Final]**:
     + **1× “market discovery”** (per event): **/events/{eventId}/markets?regions=us,us2,us\_dfs** (cost **1**). Cache available market keys per bookmaker. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
     + **Budgeted odds pulls (per event):**  
       **Every 30 minutes** in [T‑180m, T‑60m], **every 10 minutes** in [T‑60m, Final]  
       **/events/{eventId}/odds?regions=us\_dfs&markets=<curated\_list>**
       - Start with **DFS only** for props (region us\_dfs) to keep costs down.
       - Curated list (per sport, ~6 markets) from Section 1B.
       - Cost ≈ **(#unique markets returned) × 1 region** per call; e.g., **~6 credits** per event per pull. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
     + Optional: For books in us,us2, do **one snapshot per event at T‑30m** only (same curated list) to show “book” vs “DFS” contrast, not continuous 5‑min pulls (saves a lot).
4. **Scores (grading game picks):**
   * Per sport, during game windows: call **/scores?daysFrom=1** every **10 minutes**. Cost **2**/call. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
   * Persist final scores for grading moneyline/spread/totals.
5. **Historical snapshots (every 2 hours):**
   * **Featured market snapshots:** per sport:  
     **/v4/historical/sports/{sport}/odds?regions=us&markets=h2h,spreads,totals&date=<rounded-2h-UTC>**  
     Cost: **10 × 3 markets × 1 region = 30** credits per sport per run (choose us only here to save credits). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
   * **Historical props (per‑event)**: **don’t** bulk pull by default—the cost is **10×** and per event. Only fetch on demand (e.g., to reconstruct a specific slip at a specific time). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**Why this fits (rough monthly math, example month)**

* **Game odds (5‑min)**: NBA (~26 busy days) & MLB (~27) & NFL (weekly windows) → **~30–33k credits** (assumes 6 credits per call, evening windows only).
* **Historical featured (2‑hourly)**: 3 sports × 12 runs/day × 30 days × 30 credits ≈ **32,400 credits**. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Props (DFS only, curated, duty‑cycled)**
  + NBA (~10 games/day): 3 pulls/event (avg) × 6 credits ≈ 18 credits/event → ~**4.5–5k**/mo
  + MLB (~12/day): similar → ~**5–6k**/mo
  + NFL (~16/wk): ~**1–1.3k**/mo
  + **Subtotal props:** **~11–12k**/mo
* **Scores:** 2 credits per call, ~**5–6k**/mo with 10‑min cadence in game windows. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**Total**: ≈ **80–85k** credits/month (headroom for spikes).

All of the above stays below your **100k** ceiling while meeting the **5‑minute** refresh for game odds.

**4) Request shapes (Cursor can scaffold)**

Use oddsFormat=american for US display; store both **American** and **Decimal** and compute **implied probability** serverside.

**A) Game odds (league-level)**

GET /v4/sports/{sport}/odds

?apiKey=...

&regions=us,us2

&markets=h2h,spreads,totals

&oddsFormat=american

&dateFormat=iso

[&commenceTimeFrom=...&commenceTimeTo=...]

* Cost example: **3 markets × 2 regions = 6**. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**B) Per‑event props (DFS)**

GET /v4/sports/{sport}/events/{eventId}/odds

?apiKey=...

&regions=us\_dfs

&markets=player\_points,player\_rebounds,player\_assists,player\_threes,player\_points\_rebounds\_assists,player\_turnovers

&oddsFormat=american

&dateFormat=iso

* Cost example: **~6 markets × 1 region = ~6** (actual is **unique** markets returned). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**C) Scores**

GET /v4/sports/{sport}/scores?apiKey=...&daysFrom=1&dateFormat=iso

* Cost: **2** with daysFrom. Live updates ~**30s**. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**D) Historical featured snapshots (2‑hourly)**

GET /v4/historical/sports/{sport}/odds

?apiKey=...

&regions=us

&markets=h2h,spreads,totals

&date=2025-10-17T14:00:00Z

* Cost: **10 × 3 × 1 = 30** credits per sport/run. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**E) Historical event odds (on-demand reconstruction)**

GET /v4/historical/sports/{sport}/events/{eventId}/odds

?apiKey=...

&regions=us\_dfs

&markets=player\_points,player\_rebounds,...

&date=2025-10-17T14:00:00Z

* Cost: **10 × [unique markets] × [regions]**. Use sparingly. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**5) Update intervals (for schedulers)**

* Featured (h2h/spreads/totals): **~60s pre‑match**, **~40s in‑play**
* Additional markets (props, alternates, periods): **~60s pre‑match/in‑play**
* Outrights: **~5 min** pre‑match
* Interval ramps down starting **~6 hours** pre‑game. [The Odds API](https://the-odds-api.com/sports-odds-data/update-intervals.html?utm_source=chatgpt.com)

**6) Units, odds & grading**

* **Odds formats**: oddsFormat=american|decimal. API supports both. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Featured markets schema**:
  + h2h: outcomes are team names with price (American odds)
  + spreads: outcomes include point (handicap) + price
  + totals: outcomes include point (line) + price (“Over”, “Under”) [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Player props**: outcomes typically Over / Under (or Yes/No) with point (e.g., yards, points) + price. Market keys per sport listed here. [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)
* **Implied probability (American odds):**
  + If **positive** (e.g., +150): 100 / (odds + 100)
  + If **negative** (e.g., −120): (-odds) / ((-odds) + 100)
* **Grading**
  + **Game picks**: grade via /scores (has final team scores for last 1–3 days). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
  + **Player props**: The Odds API **does not** return player stat lines; integrate a stats provider (SportsDataIO, Sportradar, etc.). Keep the interface abstract so we can swap vendors.

**7) Data model (tables)**

* sports — key, title
* events — id, sport\_key, commence\_time, home\_team, away\_team, status
* bookmakers — key, title, region
* markets — event\_id, bookmaker\_key, market\_key, last\_update
* outcomes — event\_id, bookmaker\_key, market\_key, name (e.g., Over/Under, team), point (nullable), price, last\_seen
* scores — event\_id, home\_score, away\_score, completed, last\_update
* snapshots — historical captures keyed by (sport\_key, snapshot\_time, region, market\_key) with nested bookmakers/outcomes

**Uniqueness:** (event\_id, bookmaker\_key, market\_key, outcome\_name, point).

**8) Quota & back‑off**

* Inspect response headers on every call:  
  x-requests-remaining, x-requests-used, x-requests-last. Use them to throttle when near limits. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* If **429**: exponential back‑off and jitter; “space out requests over several seconds.” [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* Budget guardrails: halt lower‑priority tasks (e.g., us,us2 props) if projected month‑to‑date > 90% of cap.

**9) Helpful discovery endpoints**

* **Sports list** (free): /v4/sports (in‑season & all=true for out‑of‑season). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Participants** (teams per sport; not team rosters): /v4/sports/{sport}/participants (cost 1). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Event markets** (what a book currently has for a game): /v4/sports/{sport}/events/{eventId}/markets (cost 1). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Bookmakers by region** (incl. DFS list and notes): reference page. [The Odds API](https://the-odds-api.com/sports-odds-data/bookmaker-apis.html)
* **Betting market keys** (all featured/additional/prop keys): reference page. [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)

**10) Cursor tasks (high level)**

1. **Client wrapper** with:
   * auth, retries, 429 back‑off, header‑based quota tracking, structured logging
2. **Schedulers**
   * gameOddsPoller(sport) — 5‑min cadence when events non‑empty
   * propsPoller(sport) — event‑aware duty cycle (30→10 min), DFS only by default
   * scoresPoller(sport) — 10‑min cadence during windows
   * historicalFeaturedPoller(sport) — 2‑hour cadence (region=us)
3. **Storage** — upsert by composite keys; archive deltas; maintain snapshot store
4. **Normalizer** — odds format conversion, implied probabilities, canonical bookmaker mapping
5. **BudgetMonitor** — project month‑end credits; shed load if needed
6. **Feature flags** — toggle regions/markets per sport; enable one‑off historical event pulls for audits

**11) Example props “curated lists”**

* **NFL:** player\_pass\_yds,player\_pass\_tds,player\_rush\_yds,player\_receptions,player\_reception\_yds,player\_anytime\_td (expand as budget allows). [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)
* **NBA:** player\_points,player\_rebounds,player\_assists,player\_threes,player\_points\_rebounds\_assists,player\_turnovers. [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)
* **MLB:** pitcher\_strikeouts,batter\_total\_bases,batter\_hits,batter\_home\_runs,batter\_runs\_scored,batter\_rbis. [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)

**12) Notes & trade‑offs**

* **5‑minute** cadence is met for **game odds**. For **props**, stricter cadences explode costs because event‑level calls are per‑market and per‑region. Duty‑cycling + **DFS‑first** strikes a good balance (and captures the “Pick6/PrizePicks” use case). [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Historical props** are available, but expensive; fetch **only on demand** to reconstruct past slips. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* If you need **player stat lines** to auto‑grade props, wire a stats feed; The Odds API’s /scores is team‑level only. [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)

**Appendix — quick links to the docs you’ll use most**

* **V4 guide (all endpoints & costs):** Odds, Event Odds, Scores, Events, Event Markets, Historical (league & per‑event) [The Odds API](https://the-odds-api.com/liveapi/guides/v4/)
* **Update intervals (how often books refresh):** [The Odds API](https://the-odds-api.com/sports-odds-data/update-intervals.html?utm_source=chatgpt.com)
* **Historical coverage windows:** featured snapshots since 2020 (10→5 min), additional markets since 2023‑05‑03 (5 min) [The Odds API+1](https://the-odds-api.com/historical-odds-data/?utm_source=chatgpt.com)
* **Bookmakers by region (incl. DFS: PrizePicks, Pick6):** [The Odds API](https://the-odds-api.com/sports-odds-data/bookmaker-apis.html)
* **Betting market keys (featured, additional, props):** [The Odds API](https://the-odds-api.com/sports-odds-data/betting-markets.html)