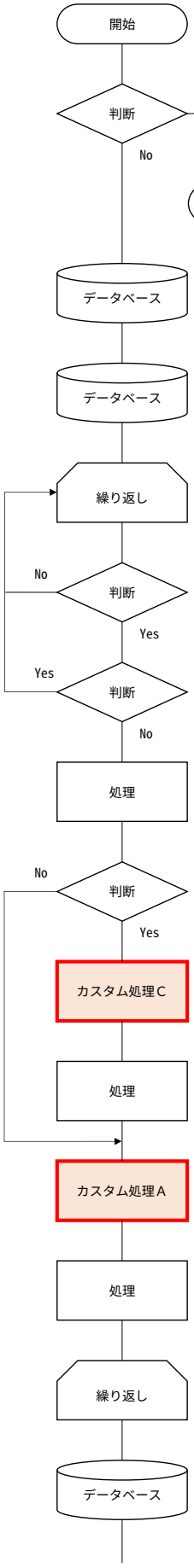
PC-KYOTEI 自動投票フローチャート



タイマーが30秒ごとに自動的に実行

前回の処理がまだ実行中？

タイマーの30秒後の実行を待つ

本日分以外の投票レースを削除します
DELETE FROM apd_tohyo_race

現在時刻 < 締切時刻 の本日分レースを取得します
SELECT * FROM brd_l2 WHERE 現在時刻 < shimekiri_yotei_jikoku

レース単位で繰り返し
※現在時刻 < 締切時刻 の本日分レースをキーに繰り返し

「自動投票実行」画面で設定した自動投票の時間(締切時刻のn分前)に到達してる？
※締切時刻がまだ先のレースはここで除外されます

投票済み？
SELECT * FROM apd_tohyo_race

レース毎の速報系データを取得します(※表1)
※「登録対象データ設定」画面で設定したデータのみ取得

(※表1)レース毎の速報系データ	
L1	出走表
01	オッズ1(単勝・複勝)
03	オッズ3(拡連複)
04	オッズ4(2連複・2連単)
05	オッズ5(3連複)
06	オッズ6(3連単)
C2	直前情報

「自動投票設定Python」画面で設定してる？
SELECT * FROM apd_tohyo_python

Pythonのデータベース･プロシージャを実行します
CALL app_custom_python
ここで出走表データを作成する自作プロシージャを呼び出します

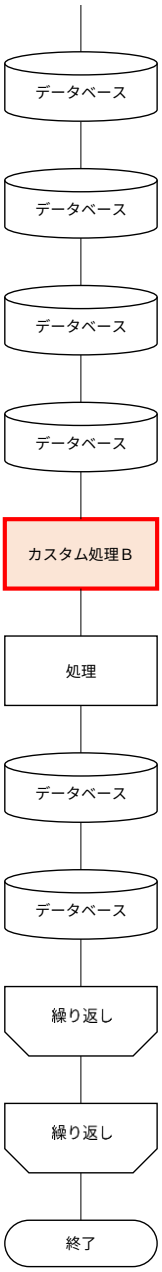
予測用ソースコード(*.py)を実行します
予測データを作成します

外部指数のデータベース･プロシージャを実行します
CALL app_custom_shisu
プロシージャで外部指数を作成(※apd_shisu_2 への書き込み)する場合は、
ここでその自作プロシージャを呼び出します

購入限度額(残高)を取得します。

投票No単位で繰り返し
※「自動投票設定一覧」画面で選択した投票Noをキーに繰り返し

「自動投票設定登録」画面の投票設定を取得します
SELECT * FROM apd_tohyo_settei



前回の買い目を一時表から削除します
```
DELETE FROM apd_kaime_tmp_1
DELETE FROM apd_kaime_tmp_2
```

今回の買い目を一時表に作成します
```
INSERT INTO apd_kaime_tmp_2
INSERT INTO apd_kaime_tmp_1
```

一時表で「自動投票設定登録」画面の内容で買い目をフィルターします
```
UPDATE apd_kaime_tmp_2
```

一時表で投票枚数を作成します
```
UPDATE apd_kaime_tmp_1
```
※「一時表」とは、一時的なデータを保存するテーブルのことです
　このシステムでは投票ファイルのデータを作成するためのテーブルです

カスタムのデータベース･プロシージャを実行します
```
CALL app_custom_procedure
```
プロシージャで投票内容を作成(※apd_kaime_tmp_* への書き込み)する場合は、
ここでその自作プロシージャを呼び出します

「BOAT RACE インターネット投票」のウェブサイトで投票を実行します
```
SELECT * FROM apd_kaime_tmp_1 INNER JOIN apd_kaime_tmp_2
    WHERE apd_kaime_tmp_2.data_kubun < '9'
```

買い目をLOGテーブルへ移動します
```
INSERT INTO apd_kaime_log_1 SELECT * FROM apd_kaime_tmp_1
INSERT INTO apd_kaime_log_2 SELECT * FROM apd_kaime_tmp_2
```

投票したレースを登録します
```
INSERT INTO apd_tohyo_race
```

投票No単位で繰り返し
※「自動投票設定一覧」画面で選択した投票Noをキーに繰り返し

レース単位で繰り返し
※現在時刻 < 締切時刻 の本日分レースをキーに繰り返し

タイマーの30秒後の実行を待つ