

شرح الـ AudioManager ومفهوم الـ AudioFocus وما يحدث وراء الستار

في ١٨ مارس

haythem.kmal

السلام عليكم أصدقائي

كعادة المواضيع ال بحب فهمها يوصل لأكثر عدد من الناس هيتم استخدام اللهجة المصرية أو يمكن بتلكك 🤔 وكالعادة برضه هتلاقي كلام كتير مكرر ومعاد فأتمنى كتر الكلام ده مايخلش تحس ان الموضوع صعب.

لو موضوع التعامل مع الـ AudioManager ده كان صعب عليك أو حاسس انك مش متمكن منه فحابب أقولك ان الموضوع سهل جدا بس بصراحة شايف ان فيه تقصير في طريقة الشرح أو يمكن بدأوا يعتمدوا علينا بزيادة وبعشم على اننا نفهم الدنيا ماشية ازاي مع نفسنا كده او يمكن لأنني بسمع الكلام لما بيقلوا حاول تطبق الحاجات دي قبل ماأشوف الشرح. المهم كل الرغي ده علشان أوصلك معلومة ان الموضوع مش صعب أوي

(ملحوظة مهمة)

الشرح ده بينطبق على الـ API بتاعها أقل من 26 يعني الموبايلات ال أقل من Android Oreo

(فكرة عامة عن الموضوع)

بما انكم بتحبوا التشبيه و الأمثلة فنقدر نقول الاتي

ان تعامل الـ AudioManager مع ال application بتاعك و الـ applications بتاع غيرك عامل زي مايكون فيه ناس في اجتماع مغلق وطبيعي فيه مسؤول أو مدير بيدير طريقة الكلام و دور كل واحد في الكلام أثناء الاجتماع ده .كمان المدير ده معاه مايك و ال معاه المايك ده معناه انه بقدر يتكلم دلوقت.

** الـ AudioManager هوا مدير الاجتماع ده وهوا ال بينظم دور كل واحد في الكلام.

** الناس ال جوه الاجتماع ده عاملين زي الـ applications بالضبط .

** المايك ده عبارة عن مفهوم اسمه الـ AudioFocus

نرجع نكمل

المهم زي ما قالك في الفيديو كده جوه الاجتماع محدش بيتكلم الا لما بيرجع للمدير الـ big boss يعني يروح يقوله أنا عاوز أتكلم بعد اذنك وساعتها المدير ده لو المايك في ايده فطبيعي يدبله المايك ويخليه يتكلم ولو أحد الأعضاء الموجودين كان معاه المايك فيباخذ المايك منه ويدبله للي عاوز يتكلم) افترض ان الناس ال جوه الاجتماع دي محترمة ومحدثش بيتكلم الا لما يطلب المايك وكمان لما حد بيطلب المايك بيتنازلوا عنه بكل سهولة للمدير علشان يوصله للشخص الثاني)

ليه أثرتنا نقطة ان الأعضاء ال جوه الاجتماع ده محترمين ؟ لأن دور مدير الاجتماع هنا مجرد التنظيم لا أكثر ولا أقل هوا كمدير بيفترض اننا محترمين وهنسمع الأوامر لكن لو عصيناها مش هيتكلم عادي هتحصل دوشة وخلاصوهوا هيفضل بيتفرج علينا.بمعنى ان الـ big boss ده ال هوا (AudioManager) مش مسيطر أوي هوا بس بيوزع عليكم لائحة تخص الكلام أثناء الاجتماع وطريقة استلام المايك ده

(AudioFocus) و بيفترض انكم ناس مثقفة و محترمة وهتمشوا تبع اللائحة دي (لائحة يعني مجموعة من الأوامر و البروتوكولات) .

يعني تقدر تقول كده لو فرضنا ان شخص معين غير محترم كان معاه المايك (AudioFocus) واحتمال كبير انه بيتكلم دلوقت أثناء الاجتماع(ماهوا معاه المايك) وجه حد غيره طلب المايك (AudioFocus) علشان يتكلم فاللائحة بتقول انه الشخص الغير محترم ده المفروض يدي المايك للمدير(AudioManager) علشان بالتبعية الشخص ال طلب المايك يعرف يستلم المايك من المدير و يتكلم لكن لأنه مش محترم فيفضل ماسك في أم المايك فاللي بيحصل ان الشخص الثاني بيروح معاه ويتكلموا في نفس المايك يعني حتى مش بيدي فرصة للشخص الثاني أنه يتكلم باحترام. كمان لو حد ماسك المايك وبيتكلم فلأنه شخص مش محترم بيروح زي العيل الغتت يتكلم معاه في نفس المايك.لك أن تتخيل منظر الاجتماع هيكون ازاي 😭 ميغة .

تمام بالضبط ده ال ممكن يحصل لو مخلص ال application بتاعك اللي بيشتغل أي ملفات صوت يتعامل مع الـ AudioManager. لو فرضنا انك عملت تطبيق بيشتغل ملفات صوت ومش يتعامل مع الـ AudioManager هيجصل الاتي :

** لو مشغل ملف موسيقى من تطبيقك وجه تطبيق تاني بيشتغل ملف موسيقى فالمفيل دول هيشغلوا مع بعض وده بيسبب انزعاج لمستخدم التطبيق ناهيك عن انك ممكن تخلي شكل التطبيقات الثانية مش حلو لأن المستخدم معيرفش الدنيا ماشية ازاي .المستخدم ممكن يفترض ان التطبيق الثاني مش عارف يقفل التطبيق الأولاني.

** لو فرضنا ان فيه تطبيق هوا ال مشغل ملف موسيقى وانت جيت بعديه وشغلت الملف بتاعك فاللثنين برضه هيشغلوا.

** شوف الثقيلة بقى لو فرضنا انك مشغل تطبيقك الروعة ده 😭 وچالك اتصال فالرنة هتفضل شغالة و ملف الموسيقى هيفضل شغال ولو جالك قلب وتفتح المكالمة هتسمع صوت الراحل ال بيكلمك مع صوت ملف الموسيقى العجيب ال انت حاطة في التطبيق ده.

نتكلم بقى ايه ال بيحصل بس بأسلوب تقني شوية

دلوقت احنا عندنا ثلاث حاجات مهمين

- 1- ال AudioManager وده عبارة عن class ووظيفته حاجتين مهمين لو طلعت بيهن من الشرح ده يبقى عملت ال عليك (1) أول وظيفة انك بتستخدم ال AudioManager في انك تعرف هل التطبيق بتاعي معاه ال AudioManager فمعرفة أشغل الملف ولا مش معايا فاسكت أحسنلي .
- (2) ثاني وظيفة زي ماقلنا هي التنظيم... بمعنى انت بتدي ال AudioManager ده ثلاث حاجات بناء عليها بينظم تعامل تطبيقك مع التطبيقات الثانية لما يحصل تداخل في تشغيل ملفات الصوت
- *أول حاجة شوية أوامر كده بتقوله فيها لو سمحت لما تاخذ مني ال AudioManager وأنا بتكلم قولني هتاخده لمدة قد ايه وبناء عليه اعمل كذا وكذا ولو فرضنا انك رجعتلي ال AudioManager قولني برضه وبناءا عليه اعمل كذا وكذا
- *ب ثاني حاجة بتقوله نوع ملف الصوت ال هتشغله علشان يعمل حسابه بس في حاجات كده ملناش دعوة بيه دلوقت.
- *ج ثالث حاجة بتقوله لو فرضنا ان فيه حاجة شغالة وأنا أخذت ال AudioManager فبعد اذنك أنا هحتاج أتكلم لمدة عبارة عن كذا وبناءا اتعامل انت مع التطبيق ال كان شغال وشوفه بيعمل ايه في الحالة دي مليش دعوة انا.
- 2- ال application بتاع حضرتك وده المفروض بيستعين بالـ AudioManager علشان ينظم طريقة تشغل ملفات الصوت اللي جواه.
- 3- ال AudioManager ده عبارة عن مفهوم يعني مفيش Object نوعه AudioManager (لاحظ اني بتكلم عن API<26) المفهوم ده شيء معنوي بس فعليا ممكن تعبر عن امتلاكه بـ Constant value . مش قلنا ان ال AudioManager ده عامل زي المايك . أهو بالنسبة للتطبيق لو معاه المايك ده (ال هوا ال AudioManager) يبقى هيكون معاه قيمة رقمية ثابتة بتعبر عن امتلاكه للمايك ده.
- من خلال الأكواد الكلام ده هيتكرر تاني وبان الله النقاط دي هتتضح أكثر .

عامة احنا كمطورين المفروض نتبع البروتوكول ده المفروض مادام فيه حاجة اسمها AudioManager بتعرف تنظم امتي ملفات الصوت تشتغل فالمفروض انه لما أحب أشغل أي ملف صوت أتعامل مع ال AudioManager وبتفترض برضه ان المطورين التانيين هيعملوا نفس الكلام علشان الدنيا تمشي ومانقش على بعض.

طيب ازاي اخلي تطبيقي يستخدم ال AudioManager class بطريقة صحيحة؟! وكمان عاوزين نعرف تطبيقنا هيتصرف ازاي في وجود تطبيقات تانية احتمال انها تشغل ملفات موسيقى في نفس الوقت؟

مبدنيا قبل ماأجوب لاحظ اني هشرح كاني بستخدم التطبيق ال قالوك اعمله بنفسك ده ال هو a mediaplayer app ال بيشتغل ملف موسيقى معين وكمان حاول تكون مرن معايا في الأكواد ال هستخدمها يعني ممكن أعمل object من كلاس معين وبعدين أعمله initiate في حين انك في الفيديو شوفتهم عملوا declaration and initiation في نفس الوقت. فكل الطرق تؤدي الى روما... أنا هحاول أضبط أكواد التطبيق ال عندي لأنه يحتوي على كمية لا بأس بها من ال Logs وهرفقه بالشرح.

تعالى نجاب أول سؤال :

تطبيق المفروض يعمل AudioManager Object ويضبطه كده ويعرفه الدنيا ماشية ازاي وأول ما تخلص تضبط ال AudioManager هوا هيتصرف بدالك في كل حاجة بس انت عرفه بس انت عاوز ايه وناوي تعمل ايه لو اخذ المايك منك(AudioFocus) من الاخر عرفه كل حاجة وهو هيتعامل.

طيب ازاي نضبط ال AudioManager

بما ان ال AudioManager ده كلاس عادي زي أي كلاس موجود عندك فاحنا هنعمل منه Object ونضبط ال AudioManager Object ده ونخليه يتعامل بعد كده

طيب نعمل (AudioManager reference variable) بالطريقة الاتية

AudioManager myAudioManagerObject ;

جميل كده عندنا reference variable نوعه AudioManager اسمه myAudioManagerObject

واحنا عاملينه global variable علشان اقدر اتحكم فيه في اي حنة من ال activity لكن لحد دلوقتي ال object متعملش .

تعالى بقى ننشئ ال AudioManager Object ده ونخليه يتعامل بعد كده واحنا بننشئ ال AudioManager Object ده عاوز أقوله انك هتكون مسؤول عن System Service تخص تشغيل ملفات الصوت وهو أصلا مش هيعرف يستقبل SystemService تانية مش مهيا أنه يستقبلها . ال systemService ده خدمات بيقدّمها النظام فمثلا يعني ممكن تلاقي حاجة اسمها POWER_SERVICE دي من اسمها تخليك تتحكم في ادارة ال power بتاع الموبايل . هتلاقي مثلا حاجة اسمها NOTIFICATION_SERVICE دي خدمة بيقدّمها لك النظام بحيث لما تستخدمها جوه تطبيقك تقدر تقول للمستخدم أي notification معينة . ال يهنا بقى حاجة اسمها ال AUDIO_SERVICE واكثر حاجة الخدمة دي تقدر تفيدنا بيها حاجة اسمها audio routing بمعنى انها تقدر تخلينا نتحكم في توجيه الأصوات بمعنى امتي نقفل الصوت من تطبيق ونوجهه لتطبيق تاني وماشابه)

طيب المهم ازاي أعرف ال myAudioManagerObject بتاعي انه لازم يستخدم الخدمة دي علشان هوا بيعرف يستغل امكانياتها بنعمل كده بالسطر التالي

myAudioManagerObject = (AudioManager) getSystemService(AUDIO_SERVICE);

طبيب لحد دلوقت تمام!؟

معانا أهو myAudioManagerObject خليه بتعامل بقى. يلا بما انه هوا المدير تعالى نطلب منه يدينا الـ AudioFocus ونشتغل بقى 😊 استنى بس هوا الـ myAudioManagerObject ده هيتعامل كده مع نفسه؟؟ لحد دلوقت هوا لا عارف انت ناوي تعمل ايه لما الـ AudioFocus يروح منك ولا ناوي تعمل ايه لما الـ AudioFocus يجيك ولا عارف انت لما تطلب الـ AudioFocus ده هتحتاجه قد ايه ولا عارف دنيتك ماشية ازاى أصلا. تعالى نعمل كده وأحب أبشرك بما انك مستعجل ان الحاجات دي بتديها للـ myAudioManagerObject كلها بتتعمل في نفس الوقت ال بتطلب فيه الـ AudioFocus بتعمله من خلال أمر اسمه requestAudioFocus ومن اسمه كده واضح انك بتطلب يكون معاك المايك (AudioFocus) بس تعالى نشوف الأمر ده بيطلب ايه كـ parameters

```
int requestAudioFocus (AudioManager.OnAudioFocusChangeListener l,
                      int streamType,
                      int durationHint)
```

زي ماحنا شايفين ان الأمر ده هيرجع لنا قيمة int يعني رقم وهو بالمناسبة رقم ثابت هتعرف كده بعدين المهم خيلنا نفهم الـ parameters ال بيستقبلها الأمر ده أول Paramaters ال هوا الـ AudioManager.OnAudioFocusChangeListener عبارة عن interface وانت بتعمل implementation للأسباب ال ذكرتها فوق ال هيا

أول حاجة شوية أوامر كده بتقوله فيها لو سمحت لما تاخد مني الـ AudioFocus وأنا بتكلم قولي هتاخده لمدة قد ايه وبناء عليه اعمل كذا وكذا ولو فرضنا انك رجعتلي الـ AudioFocus قولي برضه وبناء عليه اعمل كذا وكذا

هنضبط الـ interface ده بعد مانضبط ثاني وثالث parameters.

ثاني Parameter ال هوا int streamType علوزينه علشان نعرف الـ myAudioManagerObject الاتي وزى ماقلنا فوق فايدته كالاتي

ثاني حاجة بتقوله نوعية ملف الصوت ال هتشغله علشان يعمل حسابه بس في حاجات كده ملناش دعوة بيه دلوقت.

ثالث Parameter ال هوا int durationHint علوزينه علشان نعرف الـ myAudioManagerObject الاتي

ثالث حاجة بتقوله لو فرضنا ان فيه حاجة شغالة وأنا أخذت الـ AudioFocus فبعد اذنك أنا هحتاج أتكلم لمدة عبارة عن كذا وبناءا تعامل انت مع التطبيق ال كان شغال وشوفه بيعمل ايه في الحالة دي مليش دعوة انا.

طبيب تعالى نمسك كل parameter و نمرر القيمة المناسبة .

مبدنيا خلونا نسيب موضوع الـ interface ده دلوقت مؤقتا و تعالى ندي للـ myAudioManagerObject ثاني وثالث parameters لأنها حاجات سهلة وموجودة أصلا جوه الـ AudioManager class ومجهز هالك علشان تستخدمها وقت اللزوم. بالنسبة لثاني parameter ال هوا int streamType فهتلاقي ان الـ AudioManager class موفرلك شوية streamType وكلها static علشان كده بتستخدم الـ AudioManager class في الوصول ليها) وكلها عبارة عن قيم int. المهم في حالتنا دي بنمرر AudioManager.STREAM_MUSIC وقالك ان الـ myAudioManagerObject بيستخدمها في انها يحدد الـ volume بتاع ملف الصوت بتاعك. ماتركزش أوي انت ممكن من خلال تطبيقك بيعمل ايه تعرف تختار الـ StreamType المناسب.

- ثالث parameter ال هوا int durationHint وده نفس الكلام كده هتلاقي ان الـ AudioManager class موفرلك قيم ثابتة و static للحاجات دي ومجهز هالك هوا بس علوزك تمرله القيمة ال انت علوزها علشان يعرف يتعامل. القيمة دي بقى مهمة لسبب ؟ القيمة دي حضرتك بناء عليها بتخلي أي تطبيق شغال قبلك يتوقع حاجة معينة منك. حاول معايا تربط الدنيا ببعض. انت مش تطبيق شغال لوحدهك. لا هنا فيه نوع من النظام والتعاون مع الـ AudioManager ال هوا في حالتنا دي myAudioManagerObject يعني أدليك مثال سريع لو فرضنا انك ممرت هنا القيمة الاتية AudioManager.AUDIOFOCUS_GAIN فانت كده لما بتطلب الـ AudioFocus علشان تشغل الملف بتاعك كانك بتقول للـ AudioManager ياريت تبلغ التطبيق ال أنا أخذت منه الـ AudioFocus ان الـ AudioFocus ده هيطول شوية والمفروض زي ما هتعرف بعدين انه لو تطبيق متبرمج صح هتلاقيه بيقفل ملف الصوت خالص وبيفضي الـ resources لانك معرفه من خلال الـ AudioManager ان المايك مطول معاك شوية وياعالم بقى هتقعد ساعة ولا

ساعتين مشغلي الملف ده. لو فرضنا انك مررت القيمة الاتية `AudioManager.AUDIOFOCUS_GAIN_TRANSIENT` فانك كده بتقول للـ `AudioManager` ياريت يامدير تبلغ التطبيق انك أخذت منه الـ `AudioFocus` ان الـ `AudioFocus` هيقعد معايا لحظة بس وهرجعوه ثاني وساعتها بقي من المفترض ان التطبيق ال ادالك الـ `AudioFocus` ده مايستعجلش ويقفل الملف خالص لأ ممكن يوقف الملف وهو شوية كده وهرجعه ولما يرجعه بقدر يكمل.

لاحظ انك كتطبيق هتعمل نفس ال بيعمله التطبيق اللي انت أخذت منه الـ `AudioFocus` لأن احتمال حد ياخذ منك الـ `AudioFocus` وهنا بقي يجي دول الـ `interface` بس خليني أقولك بس ابما اننا هنشغل ملف موسيقى فهنحب ناخذ الـ `AudioFocus` لمدة طويلة ولذلك هنمرر القيمة دي `AudioManager.AUDIOFOCUS_GAIN`

- نيجي لدور الـ `AudioManager.OnAudioFocusChangeListener` أهو ده بقي ياريس ال احنا بنكتب فيه نتعامل ازاى لما حد ياخذ مننا المايك (`AudioFocus`) شوفت المثال ال قلناه من شوية أهو ده يعتبر تطبيق عملي للنقطة دي ... تعالى بقي نعمل احنا implementation للـ `AudioManager.OnAudioFocusChangeListener` بحيث نعرف الـ `AudioManager` احنا عاوزين نعمل ايه في مختلف الحالات . زي مانت عارف بما ان ده `interface` يبقى فيه `method` معينة محتاج يتعملها `Implementation` الميثود دي اسمها كده
- `onAudioFocusChange(int focusChange)` بس عاوزك بقي تركز في قيمة الـ `focusChange` دي .ليه بقي ؟؟ لأن قيمة الـ `focusChange` دي بتعتمد على قيمة الـ `durationHint` للتطبيق ال بيطلب الـ `AudioFocus` (مش قلنا اننا لما بنيجي نطلب `AudioFocus` بنمرر قيمة اسمها الـ `durationHint` وهي ال بتقول للـ `AudioManager` احنا عاوزين الـ `AudioFocus` لمدة قد ايه؟

المهم مبدنيا من دلوقتي تخيل ان فيه تطبيقين واحد منهم بتاعنا واللاتنين بيشتغلوا ملفات موسيقى. طبعاً انت لو قعدت تفكر مع نفسك كده ايه هيا الحالات ال ممكن تمر على تطبيقك لو مشغل ملف موسيقى و فجأة قرر المستخدم انه يشغل ملف موسيقى من التطبيق ثاني

هتلاقي نفسك محصور في أربع حالات

أول حالة : ان التطبيق ده يطلب من الـ `AudioManager` انه ياخذ الـ `AudioFocus` لمدة طويلة ومعنى كده وانت مغمض تقول يبقى اللي عمل التطبيق ده مرر قيمة الـ `durationHint` بحيث تكون `AudioManager.AUDIOFOCUS_GAIN` ساعتها بقي هتلاقي الـ `myAudioManagerObject` يروح يخلي قيمة الـ `focusChange` عبارة عن القيمة دي `AudioManager.AUDIOFOCUS_LOSS` ولو فكرت فيها هتلاقي ان الكلام ده منطقي وطبيعي يعني طبيعي لما حد ياخذ مني الـ `AudioFocus` تمررلي قيمة تحويلي باني خسرت أو فقدت الـ `AudioFocus` ده وده ال بيحصل والمفروض انت تخلي تطبيقك يتعامل صح بناء على القيمة اللي اتمررته دي يعني في حالتنا دي المفروض تحدد هتخلي تطبيقك يعمل ايه لما حد ياخذ منك الـ `AudioFocus` (النقطة دي هتتضح اكثر وأكثر لما تشوف الـ `implementation` . ثاني حالة : ان التطبيق ده يطلب من الـ `AudioManager` انه ياخذ الـ `AudioFocus` لمدة صغيرة يعني شوية كده وهرجعك الـ `AudioFocus` ثاني ومعنى كده وانت مغمض تقول يبقى اللي عمل التطبيق ده مرر قيمة الـ `durationHint` بحيث تكون `AudioManager.AUDIOFOCUS_GAIN_TRANSIENT` ساعتها بقي هتلاقي الـ `myAudioManagerObject` يروح يخلي قيمة الـ `focusChange` عبارة عن القيمة دي `AudioManager.AUDIOFOCUS_LOSS_TRANSIENT` كلام منطقي صح؟؟ ملاحظ التناغم ودور الـ `AudioManager` ؟؟

ثالث حالة : ان التطبيق ده يطلب من الـ `AudioManager` انه ياخذ الـ `AudioFocus` لمدة قصيرة بس في نفس الوقت بيقوله التطبيق انك أخذت منه الـ `AudioFocus` ممكن يشتغل بس يوطي صوته لو سمحت ومعنى كده وانت مغمض تقول يبقى اللي عمل التطبيق ده مرر قيمة الـ `durationHint` بحيث تكون `AudioManager.AUDIOFOCUS_GAIN_TRANSIENT_MAY_DUCK` ساعتها بقي هتلاقي الـ `myAudioManagerObject` يروح يخلي قيمة الـ `focusChange` عبارة عن القيمة دي `AudioManager.AUDIOFOCUS_LOSS_TRANSIENT_CAN_DUCK` أعيد ثاني وأقولك لو فكرت تفكير منطقي هتلاقي ان فيه تناغم حتى في أسماء القيم ال بيستخدموها ودي بالمناسبة حاجة المفروض كل ميرمج محترف يعملها أقصد انك لما تعمل `variable` هات اسم يدل على وظيفته قدر الامكان)



المهم في حالة رابعة بس فكك العملية مش ناقصة تشتت

حالة خاصة طيب فاضل بس أقولك قيمة للـ `focusChange` ال هيا لما الـ `AudioManager` يمررلك القيمة دي `AudioManager.AUDIOFOCUS_GAIN` لاحظ بقي ان القيمة بتتمررلك اتوماتيك كده بس امتى؟؟ فاكرك لما قلناك في بعض الحالة الثانية ان التطبيق ممكن يطلب ياخذ الـ `AudioFocus` لمدة صغيرة أي دي الحالة الوحيدة على قدر علمي ال فيها الـ `AudioManager` بياخذ الـ `AudioFocus` من التطبيق ده بعد ما يخلص المدة الصغيرة دي ويرجعها لك ثاني اتوماتيك كده من غير ماتطلبها (او يمكن بتطلبها من ورا الستار بس ظاهرياً على الأقل بيرجعها لك من غير أمر `requestAudioFocus`) المهم لما بيرجعك الـ `AudioFocus` دي اتوماتيك كده بيمررلك الـ `focusChange` بالقيمة دي `AudioManager.AUDIOFOCUS_GAIN`



تعالى بقي نربط الكلام ده بالـ `implementation` ما هو مش بيديني القيم دي علشان جمال عيوني

أكيد هو كـ `AudioManager` متوقع مني أضبط تطبيقك يتعامل ازاى في كل الحالات دي وبعدها زي ما قلنا بديله الـ `AudioManager.OnAudioFocusChangeListener` وهو معموله `Implementation` كده علشان يتعامل هو بقي احنا كده عملنا ال علينا .

الكود الاتي شرحه جوه التعليق وهو نفس الكود اللي كان موجود في المصادر تحت الفيديو بس الشرح عربي المهم

```

AudioManager.OnAudioFocusChangeListener afChangeListener =
    new AudioManager.OnAudioFocusChangeListener() {
        public void onAudioFocusChange(int focusChange) {

//focusChange ال قيمة مع التعامل
//من خلال if statement بحيث نعرف نقرر عمل ايه بناءا على القيمة
//التي مررها الـ AudioManager
        if (focusChange == AudioManager.AUDIOFOCUS_LOSS_TRANSIENT){
//راجع الحالة الثانية
//المفروض هنا بما اني عارف اني هفقد الـ AudioFocus لمدة صغيرة
//فالمفروض مثلا أعمل pause للـ mMediaPlayer بتاعي
//لوجه اتصال تليفوني مثلا

            // Pause playback because your Audio Focus was
            // temporarily stolen, but will be back soon.
            // i.e. for a phone call

        } else if (focusChange == AudioManager.AUDIOFOCUS_LOSS) {
//راجع اول حالة
//فالمفروض مثلا أعمل stop للـ mMediaPlayer بتاعي

            // Stop playback, because you lost the Audio Focus.
            // i.e. the user started some other playback app
            // Remember to unregister your controls/buttons here.
            // And release the kra – Audio Focus!
            // You're done.
        } else if (focusChange ==
            AudioManager.AUDIOFOCUS_LOSS_TRANSIENT_CAN_DUCK) {
//راجع الحالة الثالثة
//فالمفروض مثلا أوطي الـ volume بتاع mMediaPlayer بتاعي

            // Lower the volume, because something else is also
            // playing audio over you.
            // i.e. for notifications or navigation directions
            // Depending on your audio playback, you may prefer to
            // pause playback here instead. You do you.
        } else if (focusChange == AudioManager.AUDIOFOCUS_GAIN) {
//راجع الحالة الخاصة
//فالمفروض مثلا أعمل resume للـ mMediaPlayer بتاعي
//طبعاً resume دي الـ start بالنسبة للـ mMediaPlayer بتاعي
//ودي زي ماقلنا مرتبطة بثنائي حالة أو بأول شرط للـ if بالنسبة للكود ده
            // Resume playback, because you hold the Audio Focus
            // again!
            // i.e. the phone call ended or the nav directions
            // are finished
            // If you implement ducking and lower the volume, be
            // sure to return it to normal here, as well.
        }
    }
};

```

طيب تعالى بقى نمرر كل الحاجات ال قلنا عليها دي

```
int myAudioFocusRequestResult = myAudioManagerObject.requestAudioFocus(afChangeListener
```

اطن كده كل ال parameters واضحة والسطر ال فات ده هوا ال بيطلب ال AudioManager وفي نفس الوقت زي ماقلنا بنمرر ال parameters دي لأننا لما بنطلب ال AudioManager فال AudioManager أو بمعنى أصح ال myAudioManagerObject يحتاج يعرف تطبيقك هيتعامل ازاى زي ماقلناك فوق كده.

بعد ما بنطلب ال AudioManager فانت هتستقبل احدى قيمتين ياريس والله لو ال AudioManager عرف يجيبك ال AudioManager هتلاقي قيمة ال myAudioFocusRequestResult عبارة عن AudioManager.AUDIOFOCUS_REQUEST_GRANTED وده معناه انه معاك ال AudioManager وانت كده بتطبع النظام وشغل ملفك براحتك أو يخليهاك AudioManager.AUDIOFOCUS_REQUEST_FAILED وده معناه انه معرفش يجيبك ال AudioManager وانت لو تطبيق محترم مش هتتشغل ملف الموسيقى بتاعك. افتكر قلناك ايه فوق بخصوص ال AudioManager

haythem.kmal:

الـ AudioManager ده عبارة عن مفهوم يعني مفيش Object نوعه AudioManager (لاحظ انى بتكلم عن API<26) المفهوم ده شىء معنوي بس فعليا ممكن تعبر عن امتلاكه بـ Constant value . مش قلنا ان الـ AudioManager ده عامل زي المايك . أهو بالنسبة للتطبيق لو معاه المايك ده (ال هوا الـ AudioManager) يبقى هيكون معاه قيمة رقمية ثابتة بتعبر عن امتلاكه للمايك ده.

اهو القيمتين دول بيعبروا عن مفهوم ال AudioManager أو عن امتلاكه بمعنى أصح .تمام كده؟

حاجة أخيرة يفضل انك تعملها وهي انك وقت ماتكون عارف ان خلاص خلصت تشغيل ملف الموسيقى فيستحسن انك تتخلص من المايك (الـ AudioManager) في الفيديو وضعوا الجملة دي في () releaseMediaPlayer ال هوا ادي فرصة لغيرك يا أخي 😊 حابب بس أكرر ان الكلام ده بينطبق على الموبايلات الـ API<21 المهم السطر ده كانه بيقول للـ AudioManager خلاص انا مش عاوز المايك تاني وبالتالي فكك من الأوامر ال كنت قايلك عليها في الـ afChangeListener لأنها منطقيا كده بتعتمد على تداول المايك (الـ AudioManager) بين تطبيقى والتطبيقات التانية

myAudioManagerObject.abandonAudioFocus(afChangeListener);

طيب بالنسبة لأماكن وضع كل الاكواد ال فانت دي فأعتقد انك لو فهمت الدنيا ماشية ازاى هتعرف تتعامل وعامة هما شرحوا النقطة دي بس انت خلاص مادام عرفت كل سطر بيعمل ايه هتعرف المكان المناسب . ليه بقولك كده ؟؟ لأنك ممكن عن طريق وضع سطر معين في مكان معين تخلي برنامجك يمشي في اتجاه مختلف عن الطبيعي . هنفهم كلامي أكثر لو جربت التجربة الجاية دي 😊

بالنسبة للسؤال اللي بيقول "عاوزين نعرف تطبيقنا هيتصرف ازاى في وجود تطبيقات تانية احتمال انها تشغل ملفات موسيقى في نفس الوقت ؟" هتلاقي اجابته في الموضوع ده

تكملة لموضوع ال AudioManager ومفهوم ال AudioManager

شرح الـ AudioManager ومفهوم الـ AudioManager وما يحدث وراء الستار Continuing the discussion from : محتاج تروح للموضوع الأول علشان تتابع معانا 😊 لقيت أفضل طريقة لاجابة السؤال ده انه أقولك اعمل التجربة الاتية وماتكسلش ياريت لان دي أول حاجة عملتها علشان أفهم الدنيا ماشية ازاى . (بعض الملحوظات الخاصة بالمشروع) ** احنا هنعمل برنامج بيشتغل ملف raw وتحط اسم ملف الموسيقى ال موجود عندك في ال lonely موسيقى معين فانت أكيد هتغير اسم ملف الموسيقى ال هوا هنا اسمه folder ... وياريت يكون ملف مدته

(حضرناك أنا عديت حاجة اسمها body limit) 🤖

فاضطريت اعمله في موضوع جديد.

كده خلصنا

حابب بس أقولك بالنسبة للتجربة التانية دي لو موصلتش من أول مرة فده طبيعي ولاحظ انى كتبت الكلام ده كله بعد محاولات كتير يعني مفيش حاجة بتفهم من أول مرة فماتخليش كتر الكلام يحسك ان الموضوع صعب .بالعكس انت لو عديت الأكواد ال احنا كتبناها وتخص الموضوع ده هتلاقيهم ست سبع سطور بالكثير.

تحياتي وبالتوفيق في الامتحان يارب ليننا كلنا